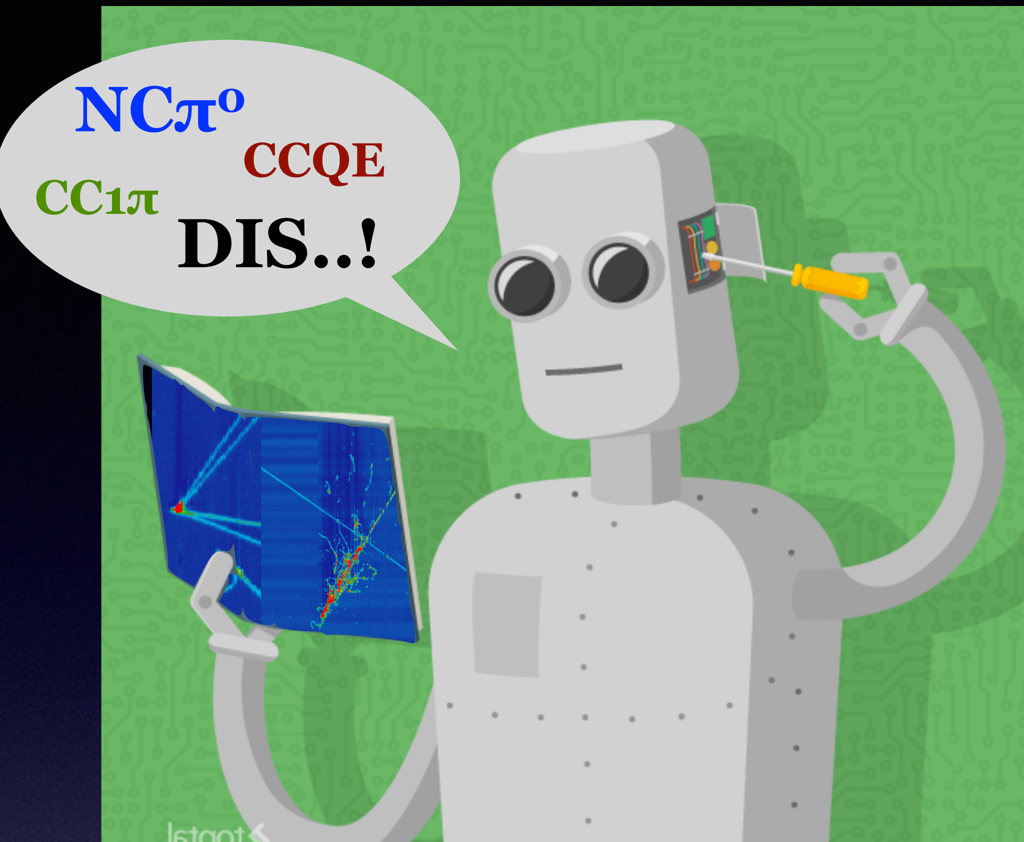
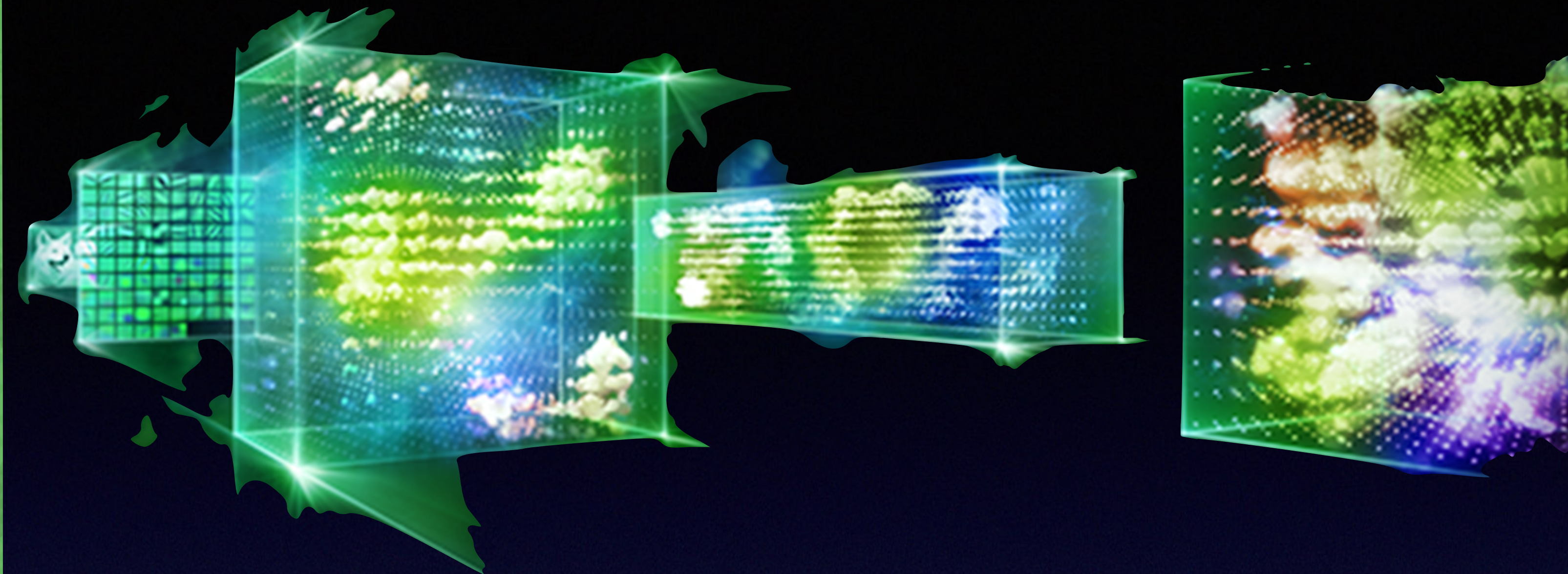


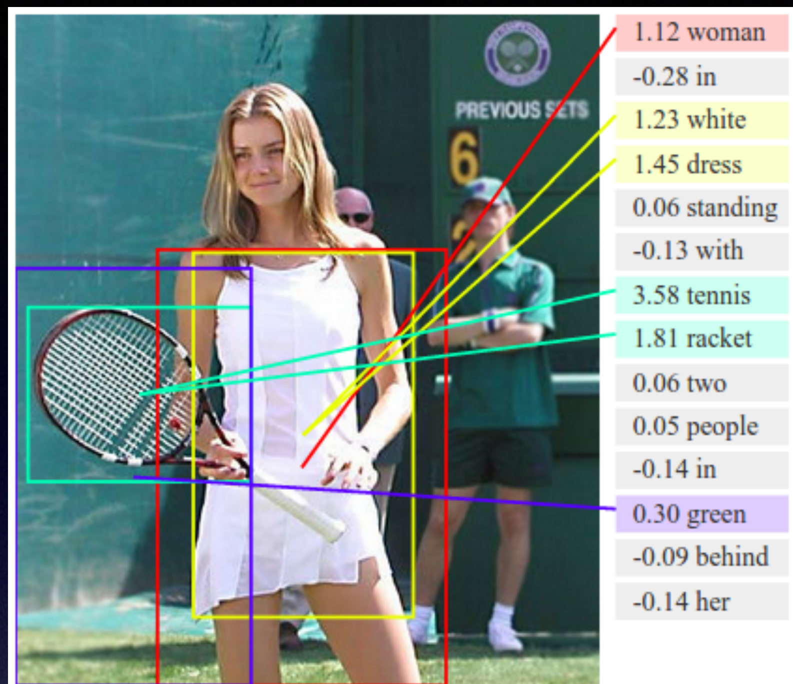


Introduction to Convolutional Neural Networks for Homogeneous Neutrino Detectors

Outline

1. **Introduction: naive words on how CNN works**
2. Image analysis applications
3. Summary

Introduction to CNNs (I)



Context Detection

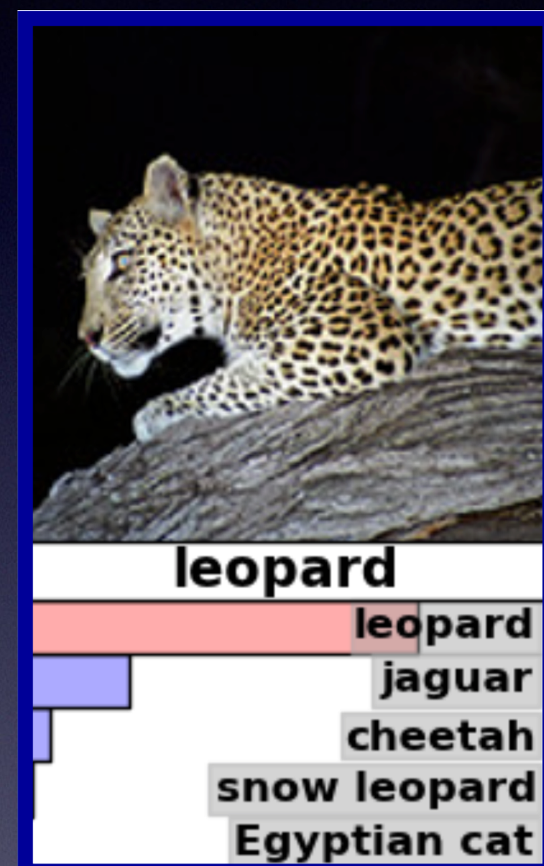
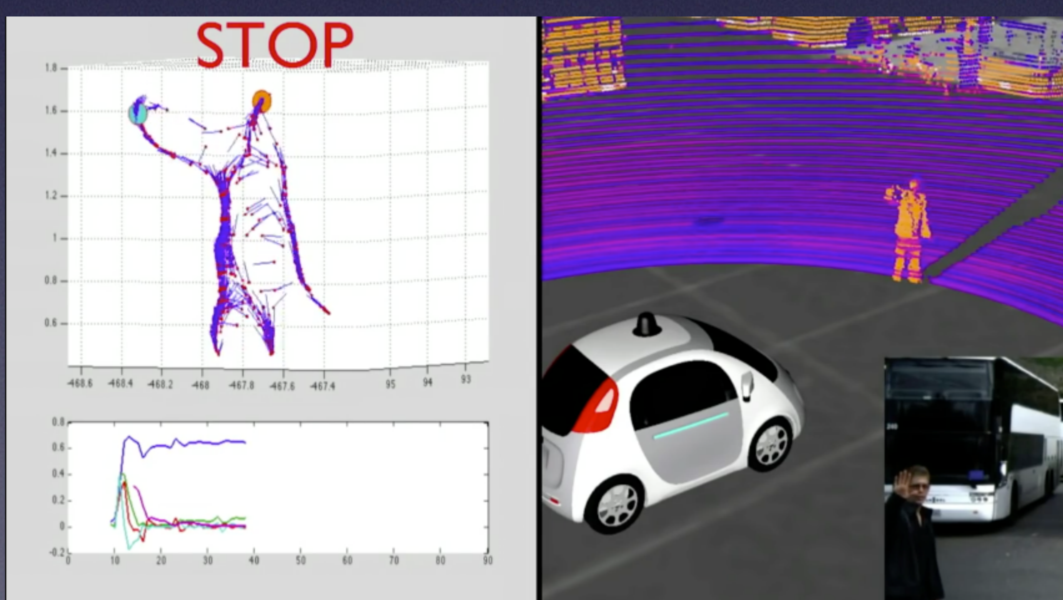


Image Classification

Image captioning



Human pose analysis (self-driving car & police)

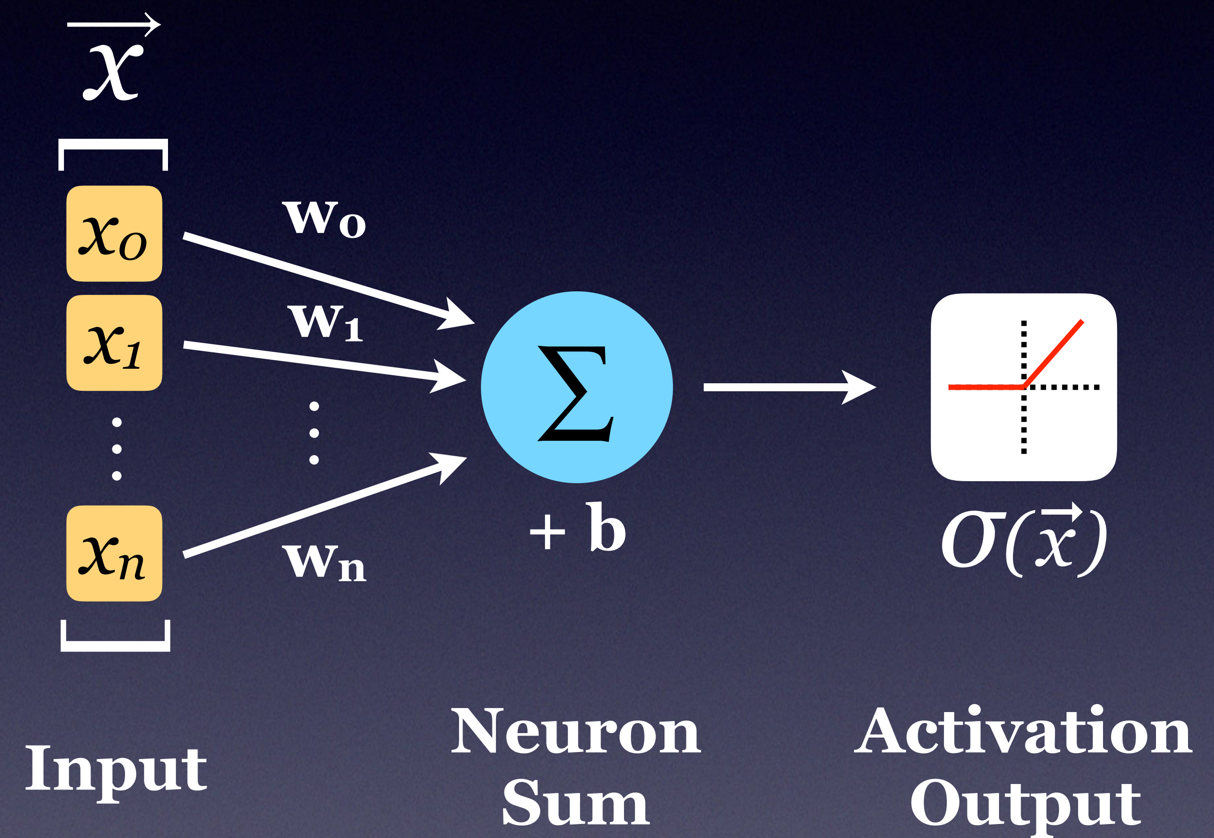
Pixel Classification

Introduction to CNNs (II)

Background: Neural Net

The basic unit of a neural net is the ***perceptron*** (loosely based on a real neuron)

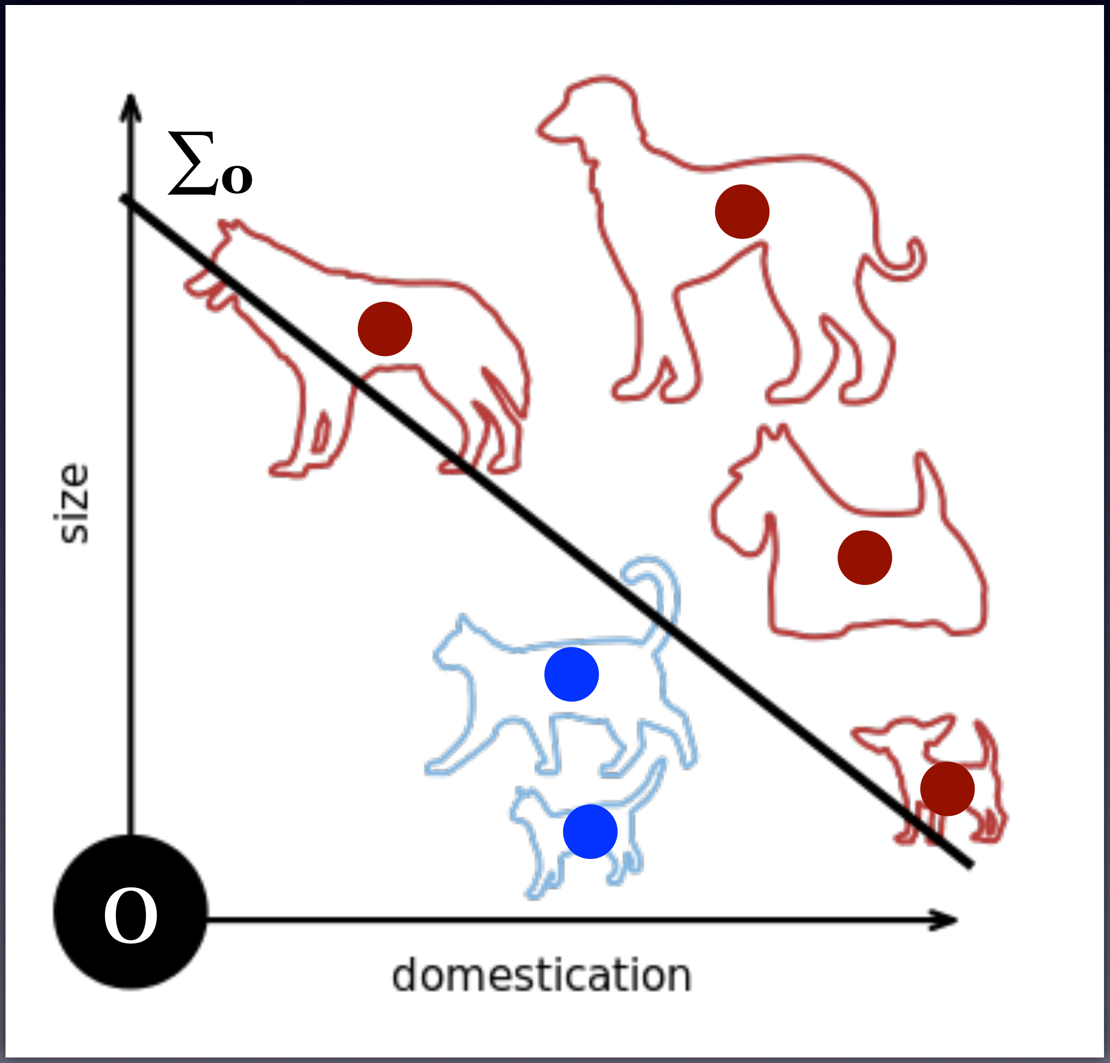
Takes in a vector of inputs (x). Commonly inputs are summed with weights (w) and offset (b) then run through activation.



$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$

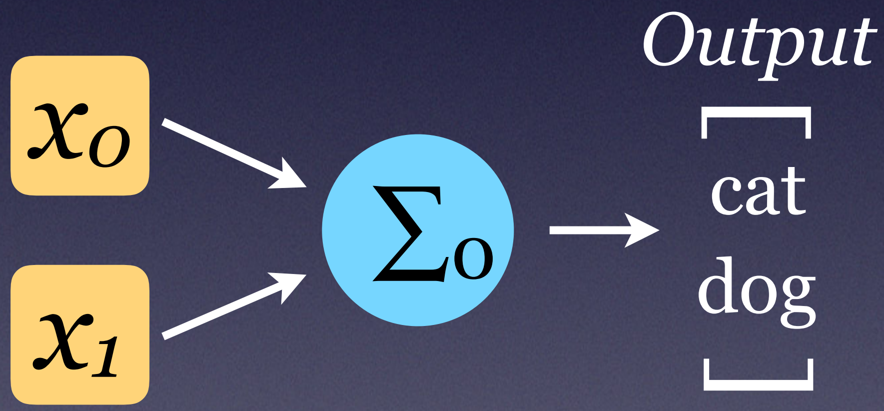
Introduction to CNNs (II)
Perceptron 2D Classification

Imagine using two features to separate cats and dogs



from wikipedia

$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$

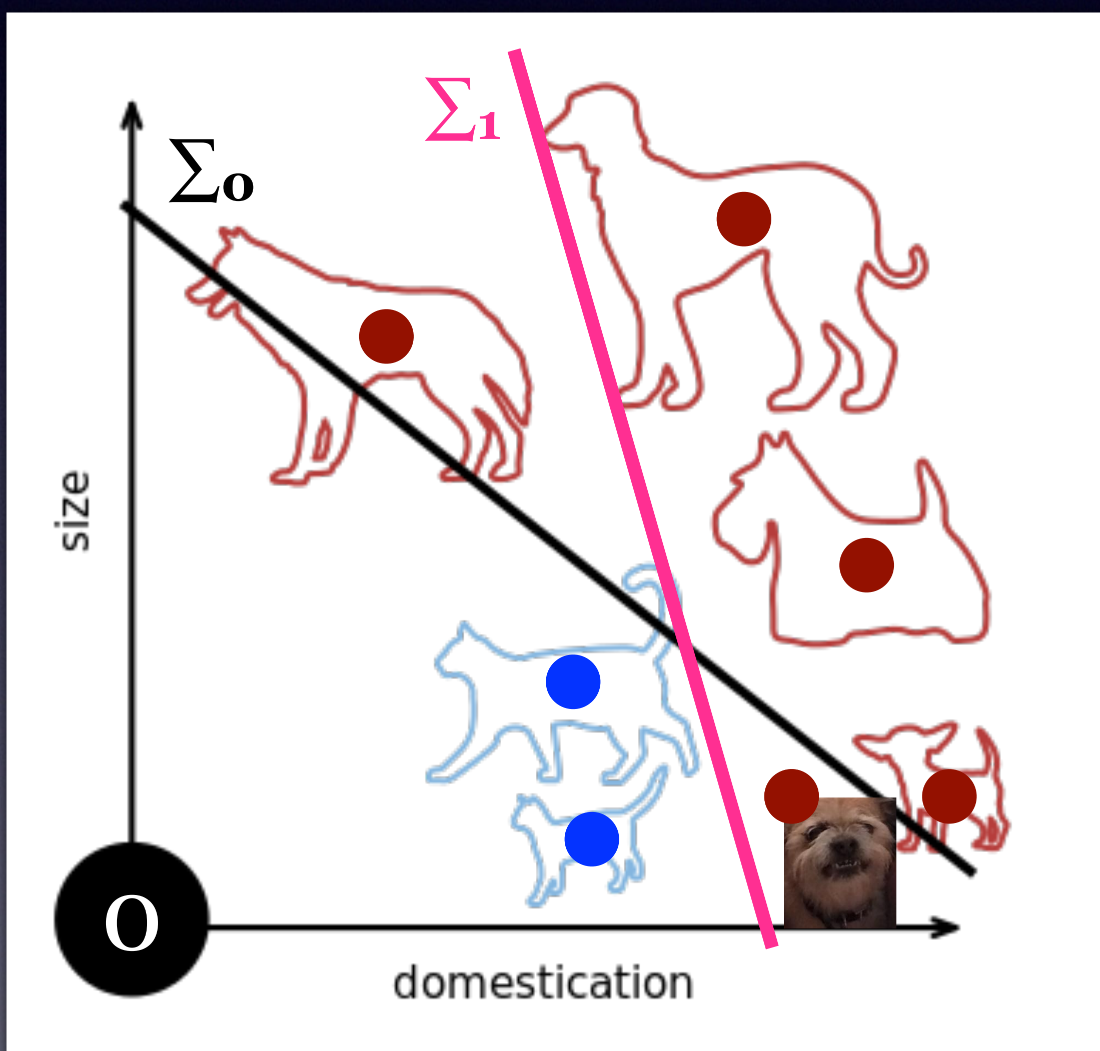


By picking a value for w and b, we define a boundary between the two sets of data

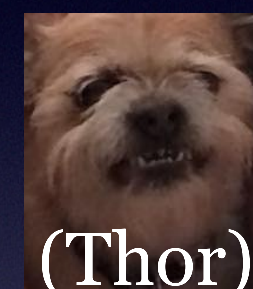
Introduction to CNNs (II)

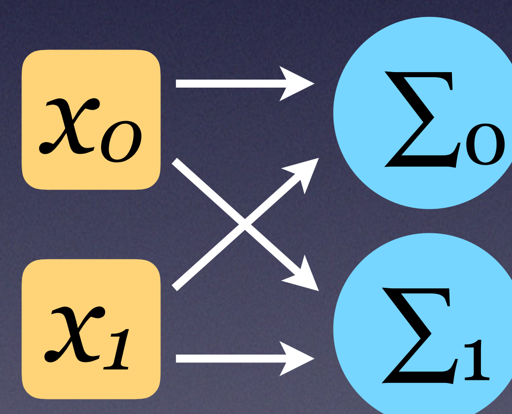
Perceptron 2D Classification

Maybe we need to do better: assume new data point
(My friend's dog — small but not as well behaved)



from [wikipedia]()



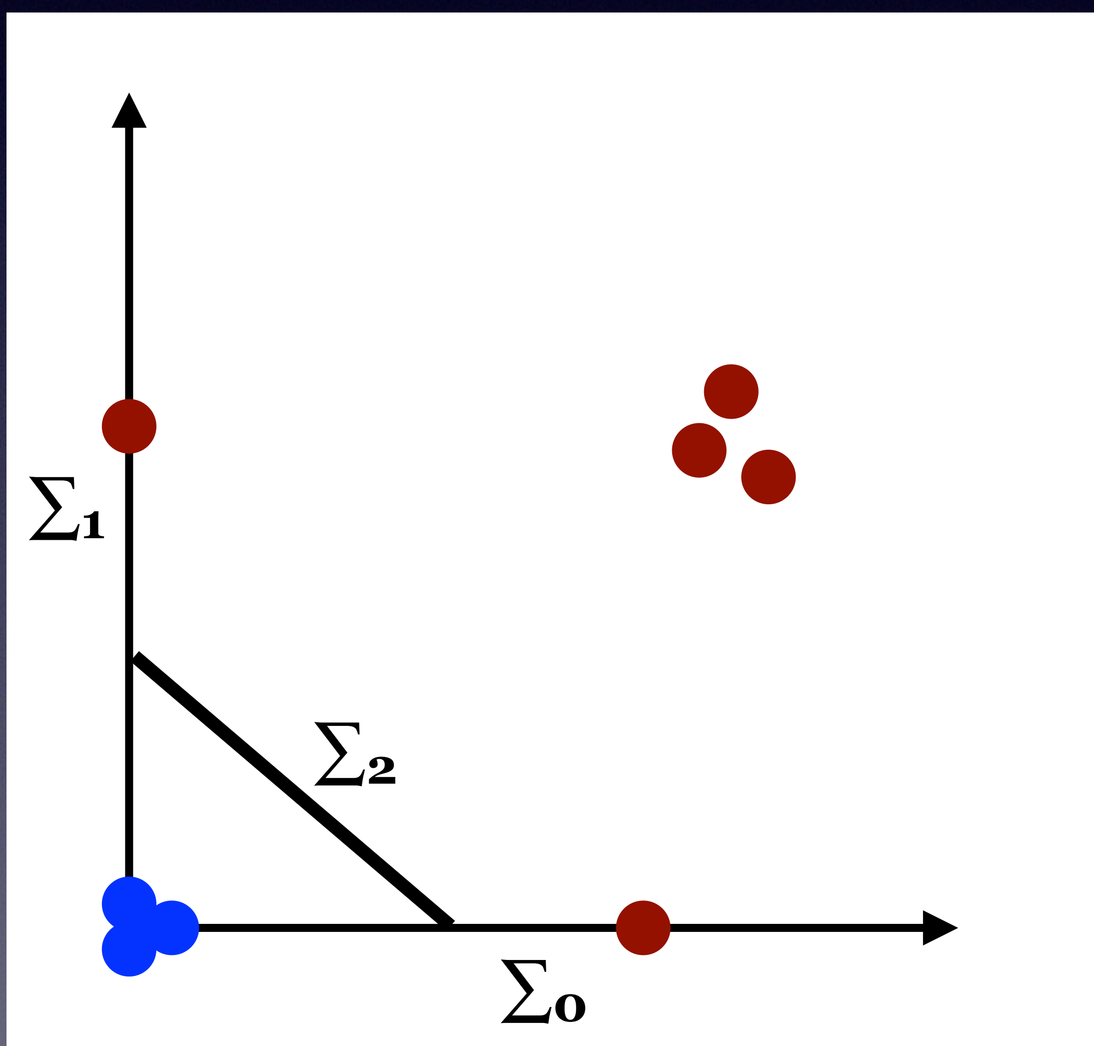


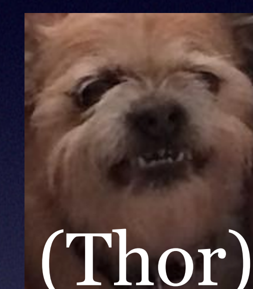
We can add another perceptron
to help classify better

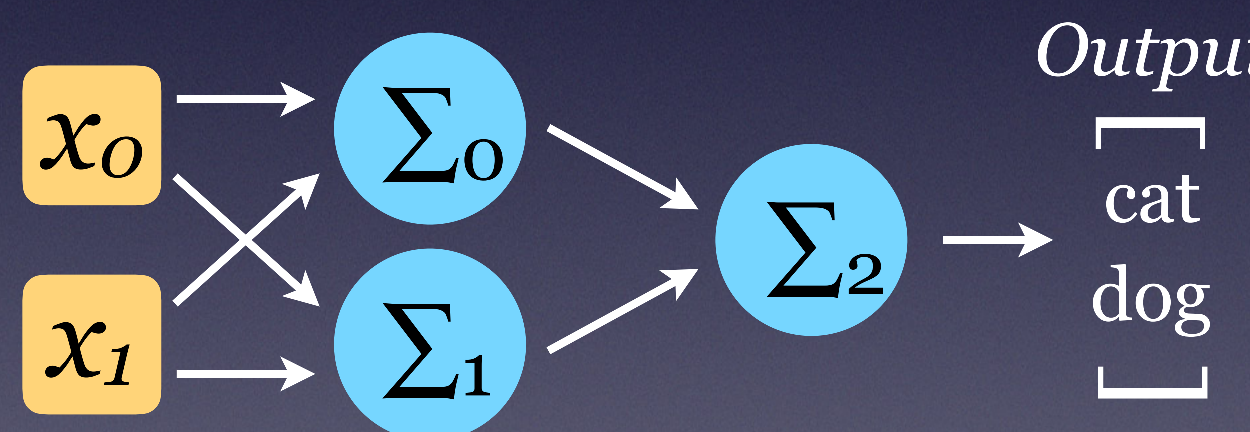
Introduction to CNNs (II)

Perceptron 2D Classification

Maybe we need to do better: assume new data point (My friend's dog — small but not as well behaved)





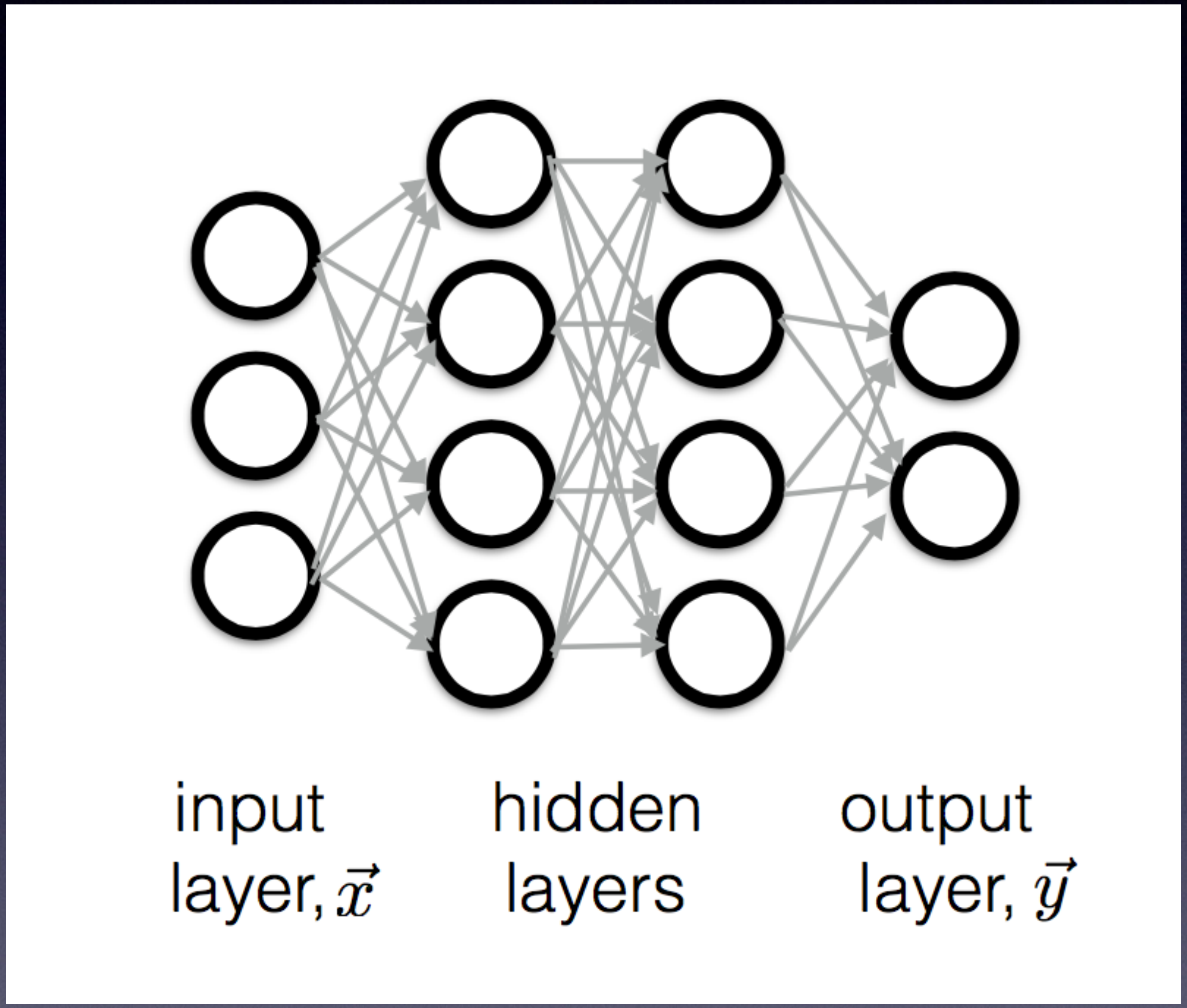


Another layer can classify based on preceding feature layer output

Introduction to CNNs (III)

"Traditional neural net" in HEP

Fully-Connected Multi-Layer Perceptrons



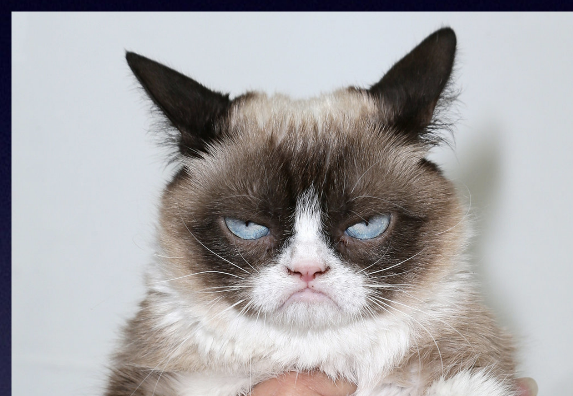
A traditional neural network consists of a stack of layers of such neurons where each neuron is ***fully connected*** to other neurons of the neighbor layers

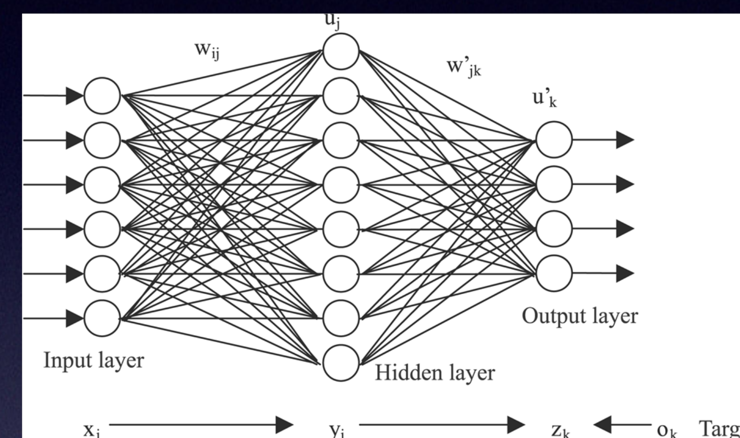
Introduction to CNNs (III)

“Traditional neural net” in HEP

Problems with it...

Feed in entire image

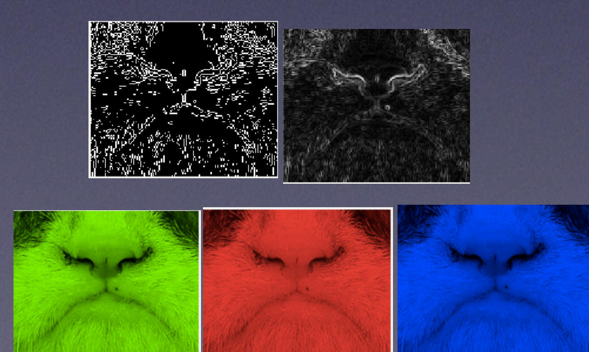


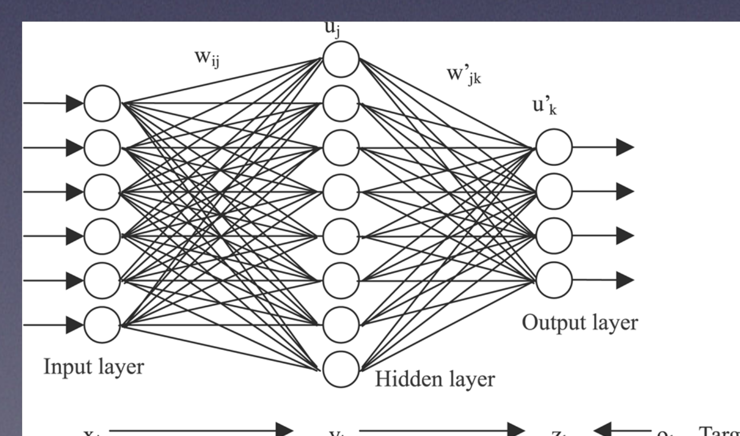


Cat?

Problem: scalability

Use pre-determined features



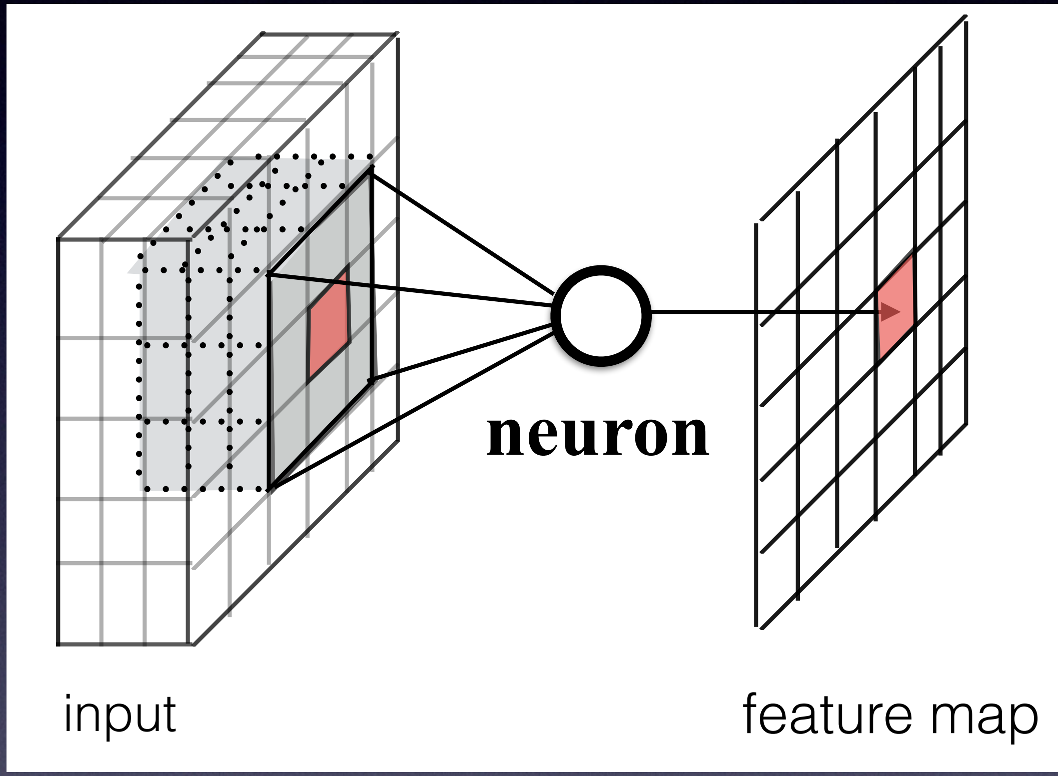


Cat?

Problem: generalization

Introduction to CNNs (III)

CNN introduce a ***limitation*** by forcing the network to look at only local, translation invariant features

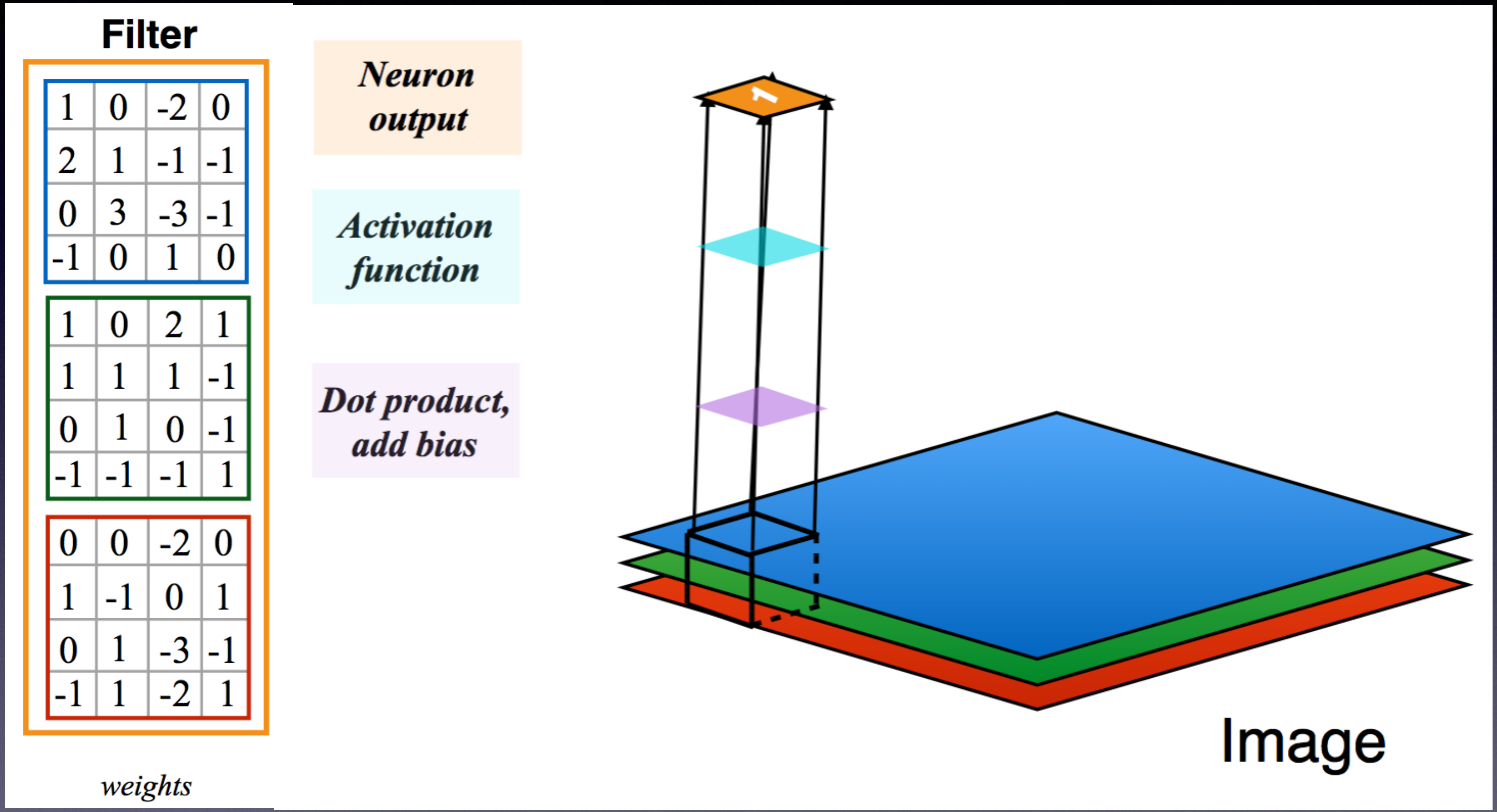


$$f_{i,j}(X) = \sigma\left(W_i \cdot X_j + b_i\right),$$

Activation of a neuron depends on the element-wise product of 3D weight tensor with 3D input data and a bias term

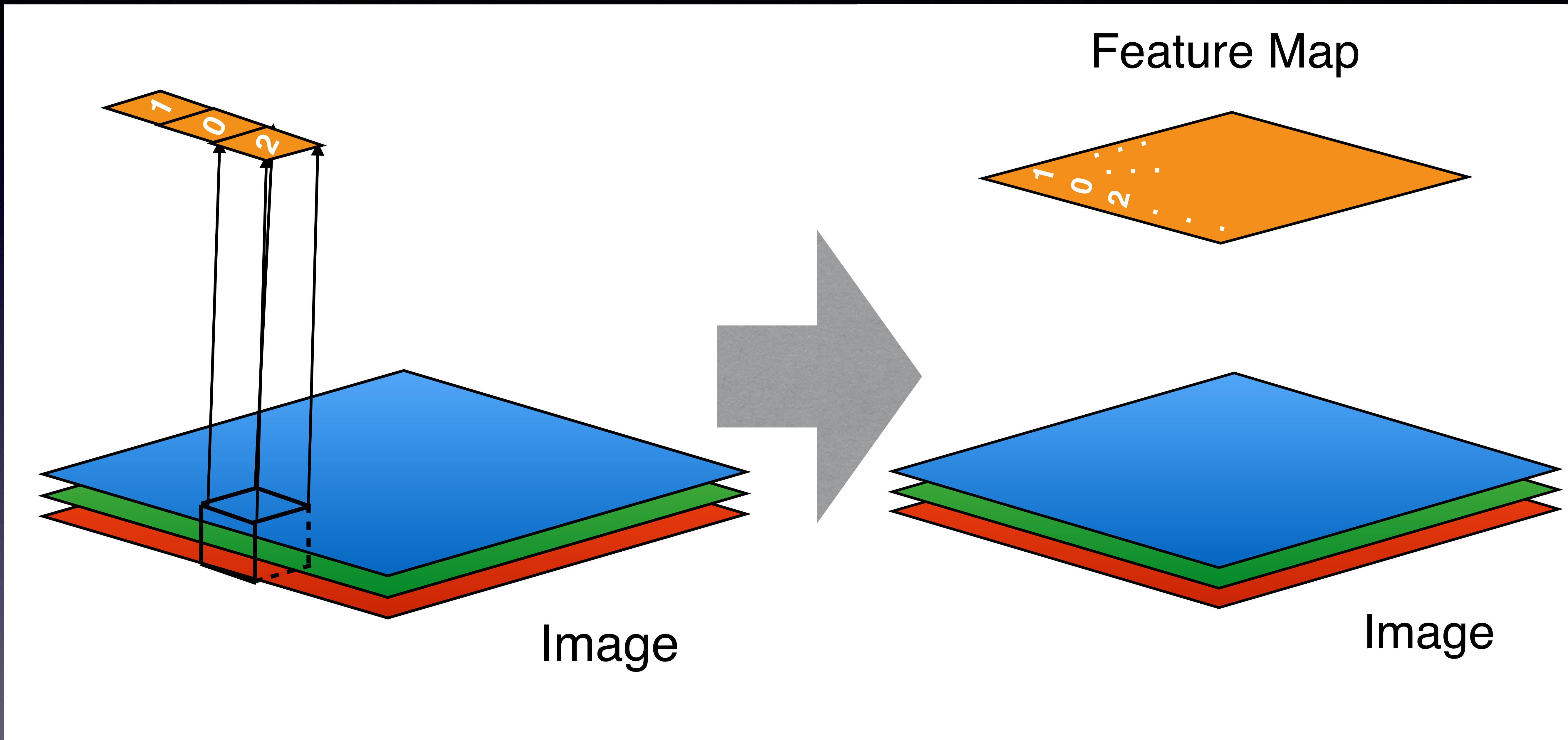
- Translate over 2D space to process the whole input
- Neuron **learns translation-invariant features**
  - Suited for a "***homogeneous***" detector like LArTPC
- **Output**: a "feature-enhanced" image (***feature map***)

Introduction to CNNs (III)



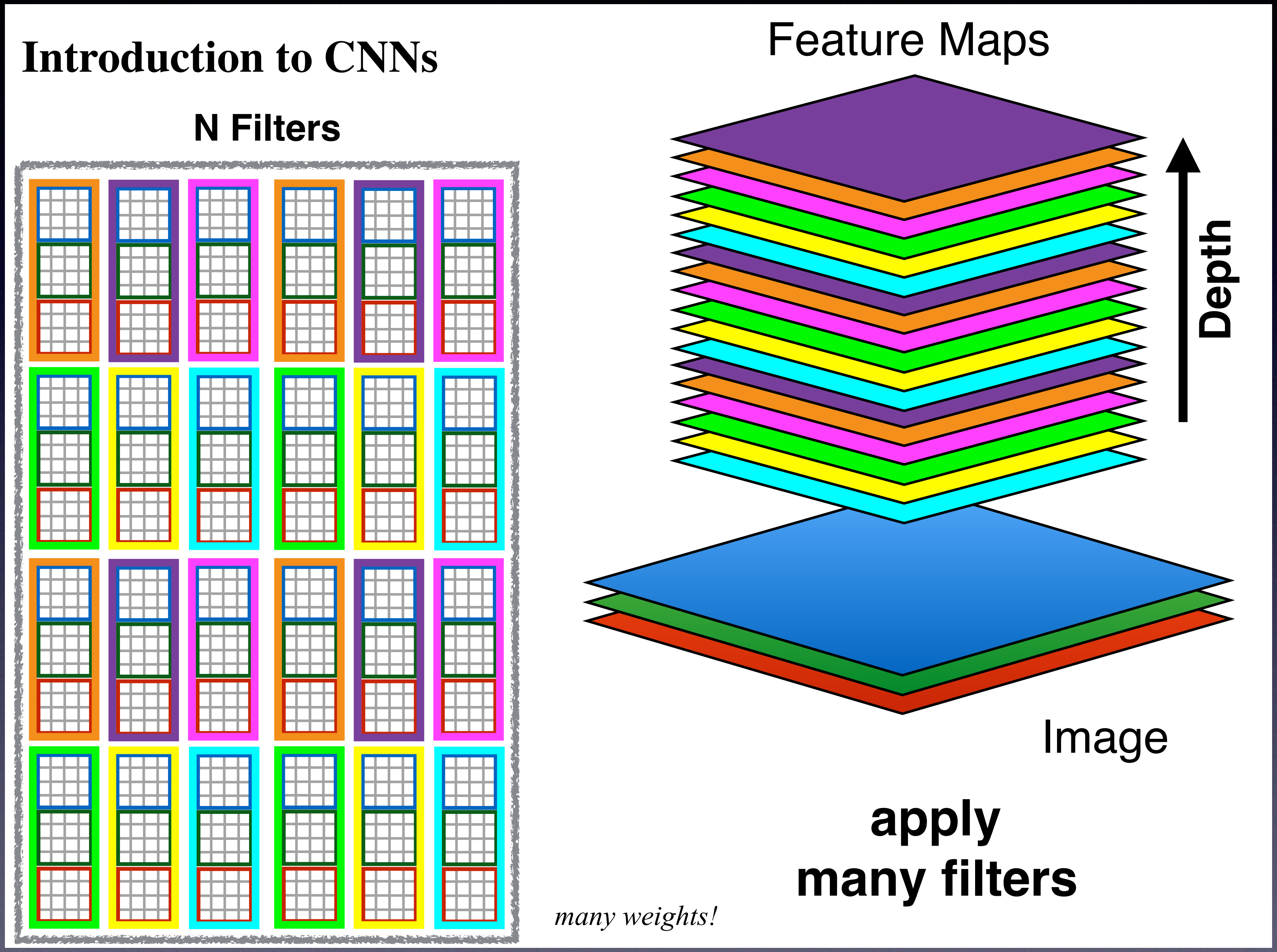
Toy visualization of the CNN operation

Introduction to CNNs (III)



Toy visualization of the CNN operation

Introduction to CNNs (III)

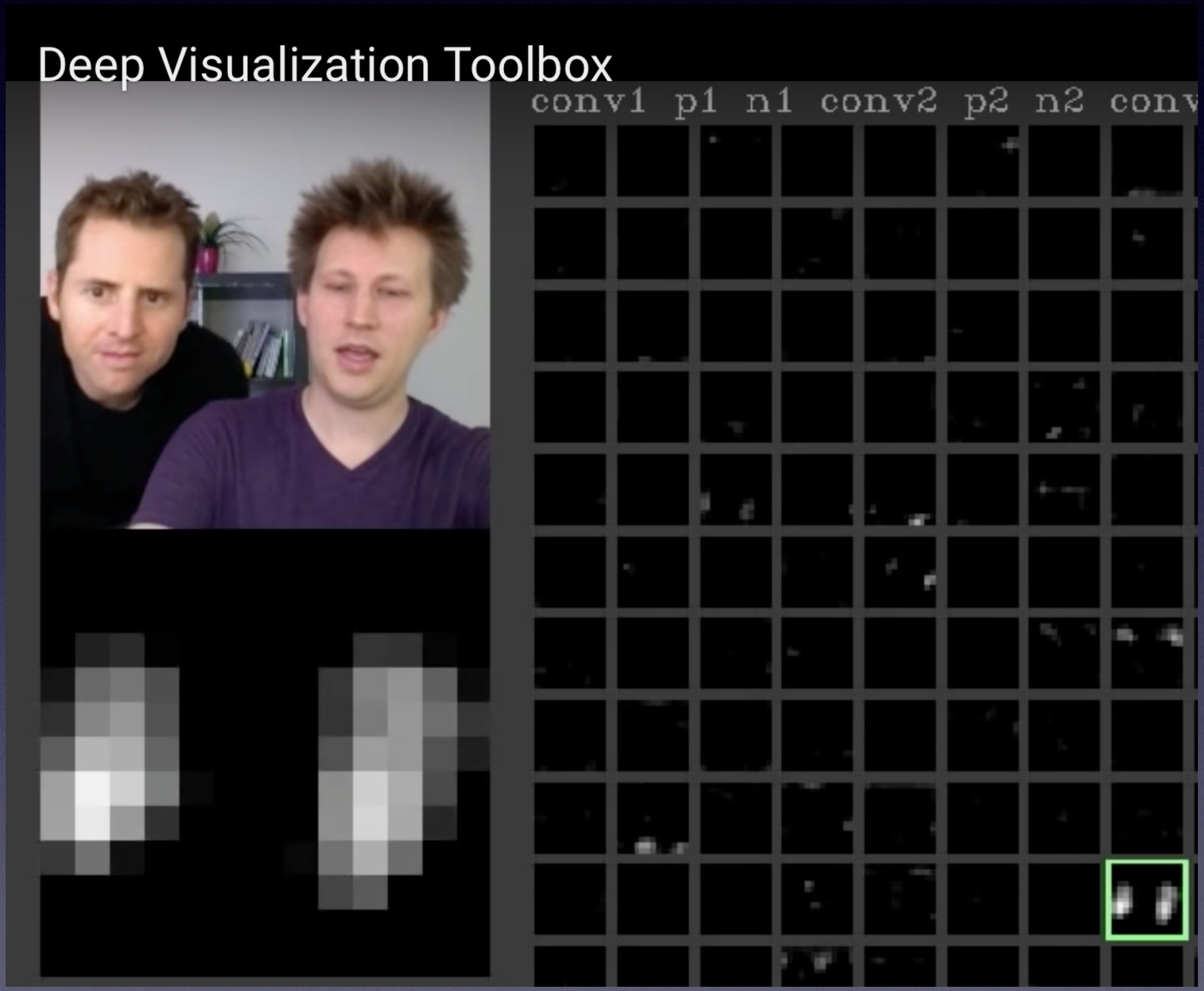


Toy visualization of the CNN operation

Introduction to CNNs (III)

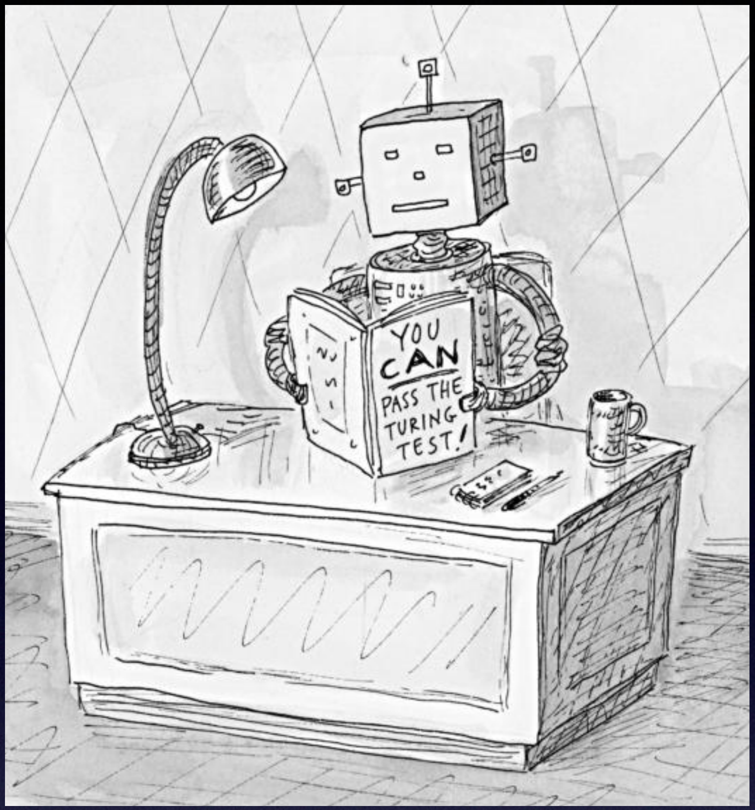
Feature map visualization example

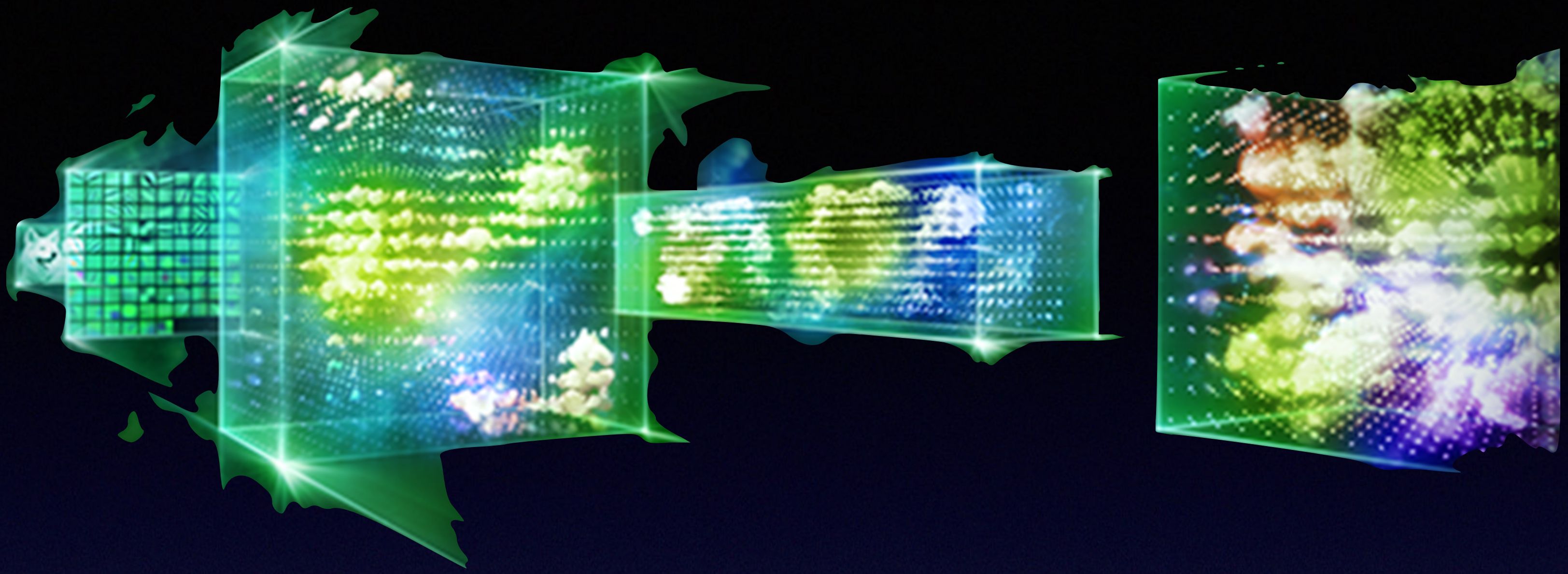
- https://www.youtube.com/watch?v=AgkfIQ4IGaM



Neuron concerning face

Neuron loving texts





Introduction to Convolutional Neural Networks for Homogeneous Neutrino Detectors

Outline

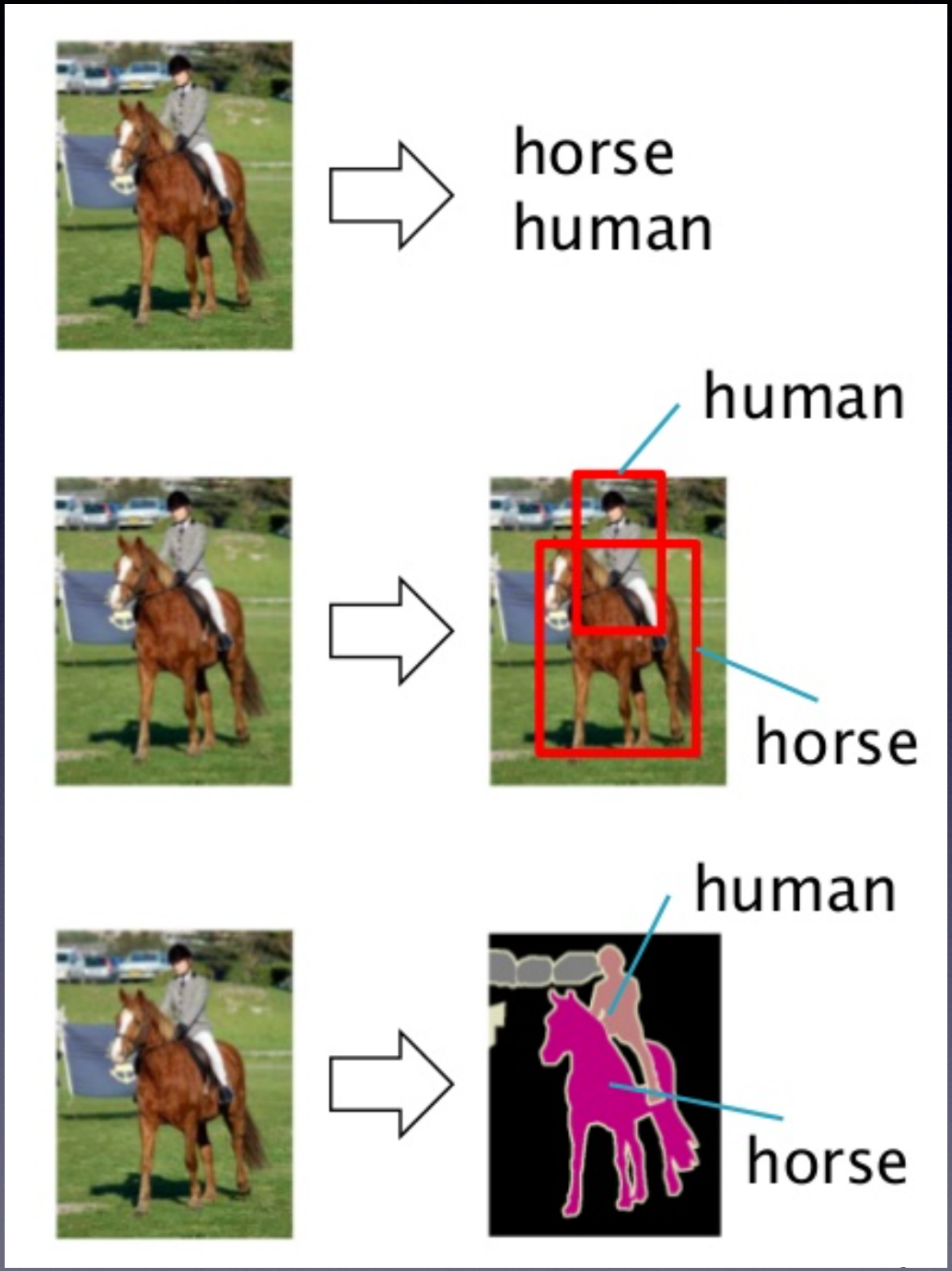
1. Introduction: naive words on how CNN works
2. **Image analysis applications**
3. Summary

Application of CNNs

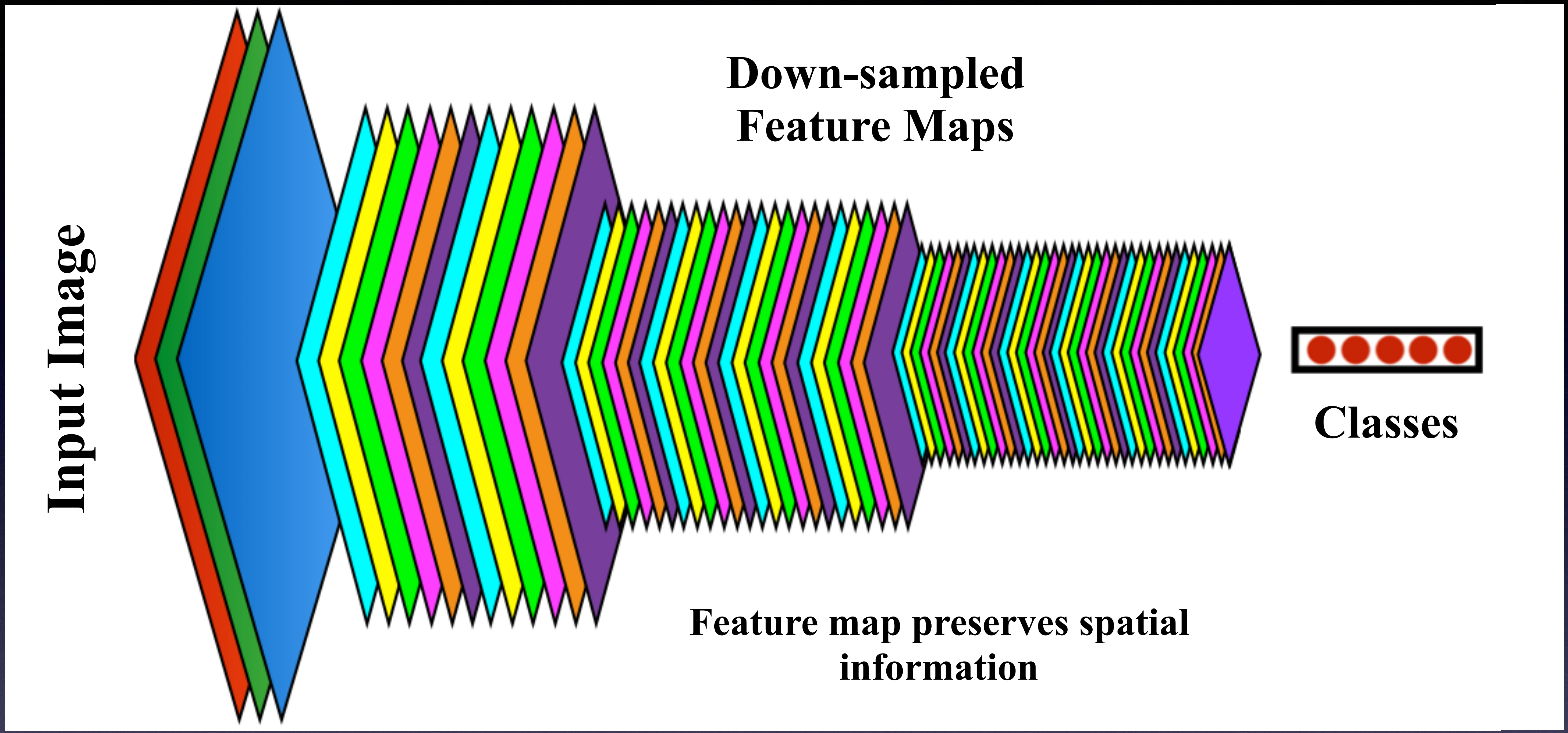
- **Categorization**
  - What's in a picture?
  - Particle ID

- **Detection**
  - What in where? (bounding box)
  - Find a neutrino

- **Semantic Segmentation**
  - WHAT IN WHERE (pixel level)
  - Clustering!



CNN for Image Classification



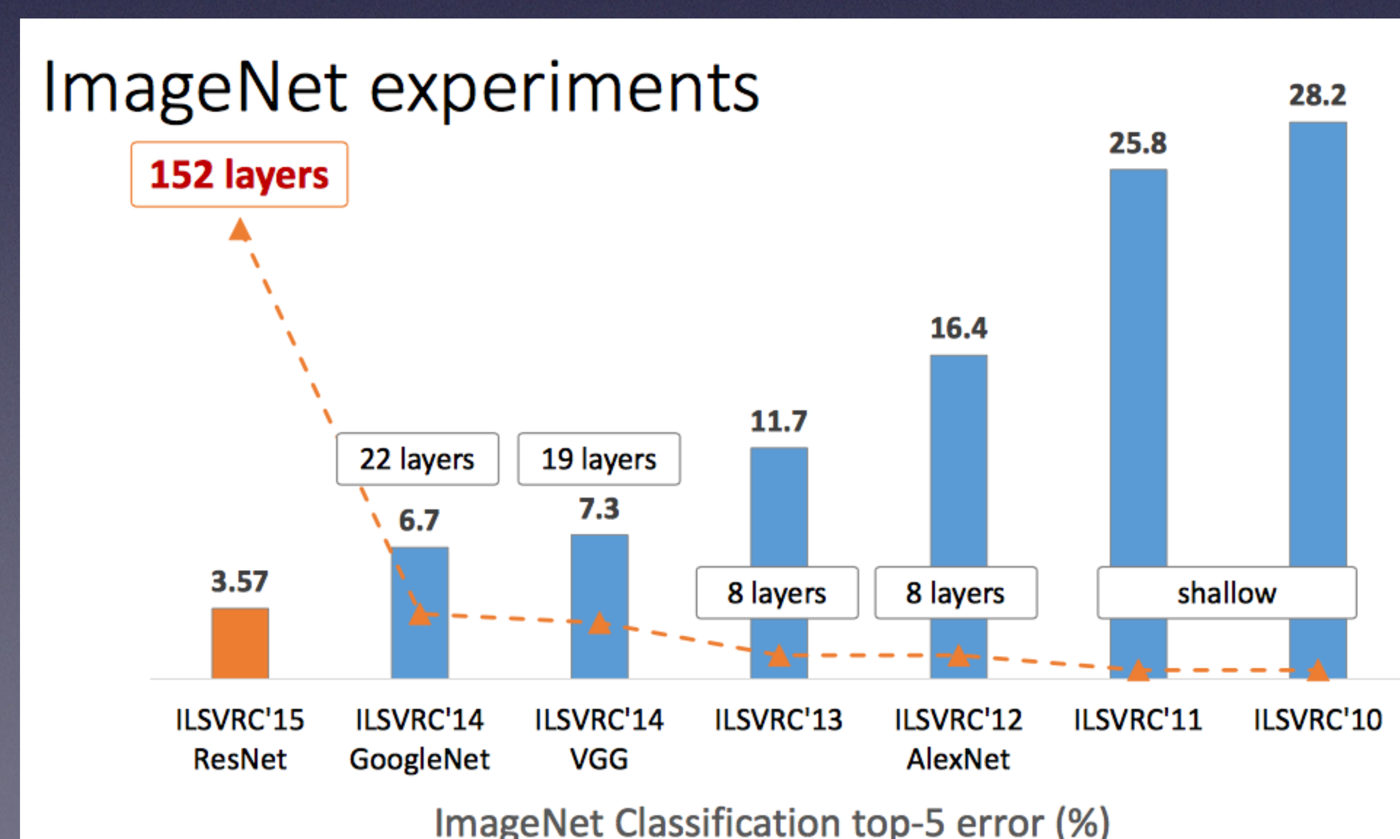
- **Goal**: provide a single label for the whole image
- **How**: transform the higher spatial resolution input (i.e. image) into a vector of image features, ultimately a 1D array of feature parameters, useful for image classification

CNN for Image Classification

ImageNet: Large Scale Visual Recognition Challenge

- ImageNet holds large image database
  - 14,000,000 pictures 22,000 categories
- ILSVRC: competition!
  - 1000 class categorization
    - 1200000 training images
    - 50000 validation, 100000 testing

Husky vs. Eskimo Dogs
(classification)



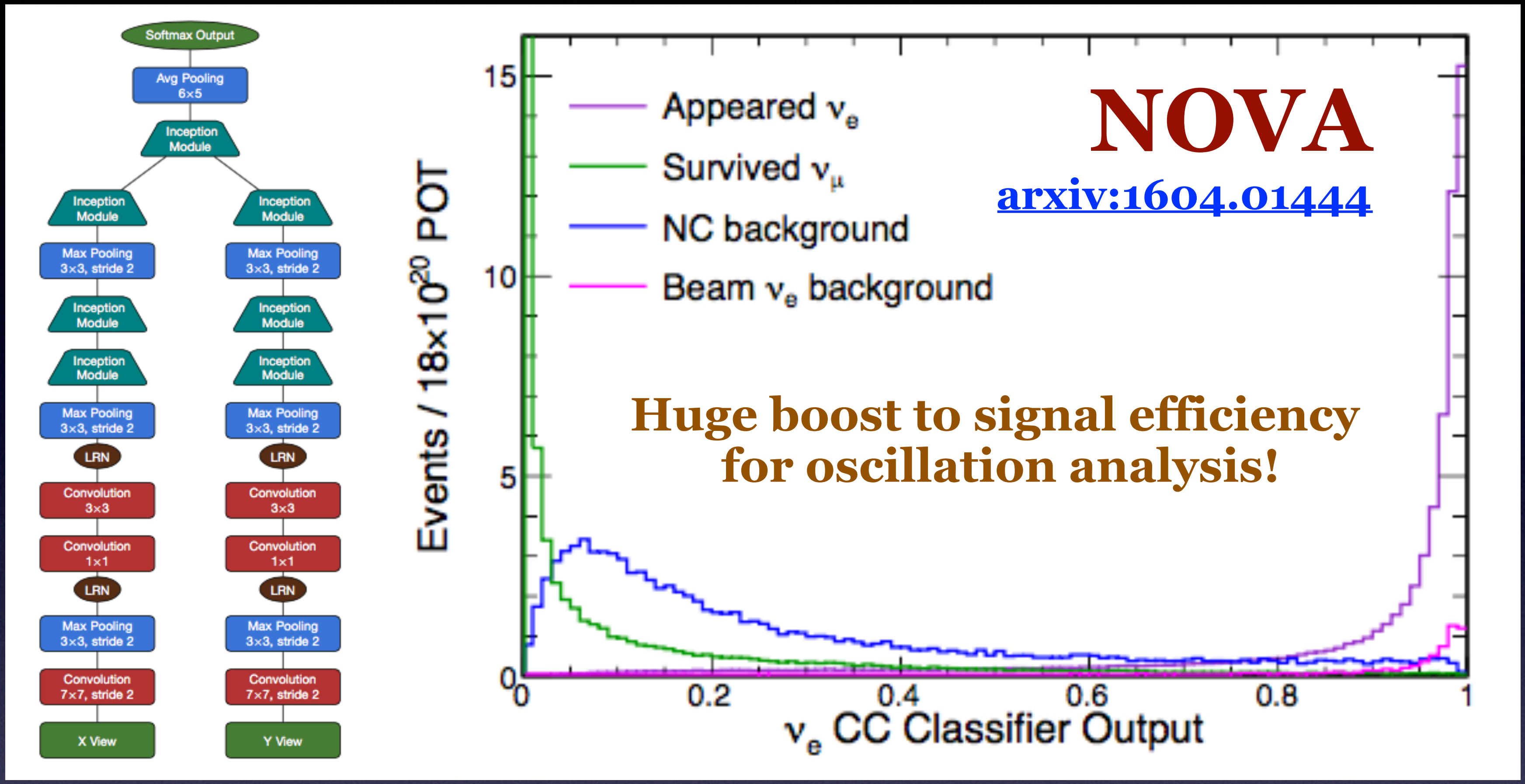
CNN for Image Classification

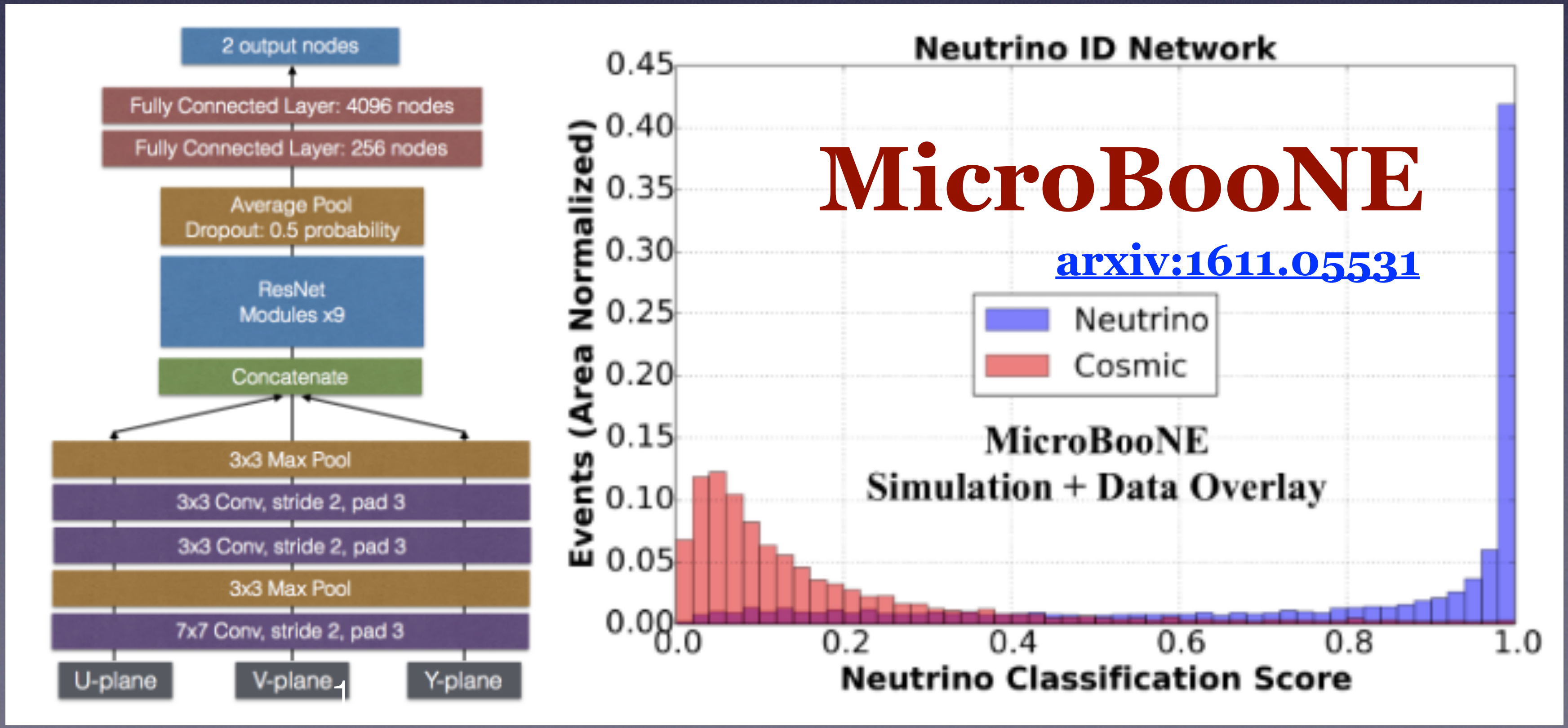
Neutrino Event Classifier

Nova & MicroBooNE both homogeneous neutrino detectors

Neutrino event classifier using 2D projection images

"Siamese Tower" Feature abstraction (spatial contraction) per plane first, then concatenate feature maps



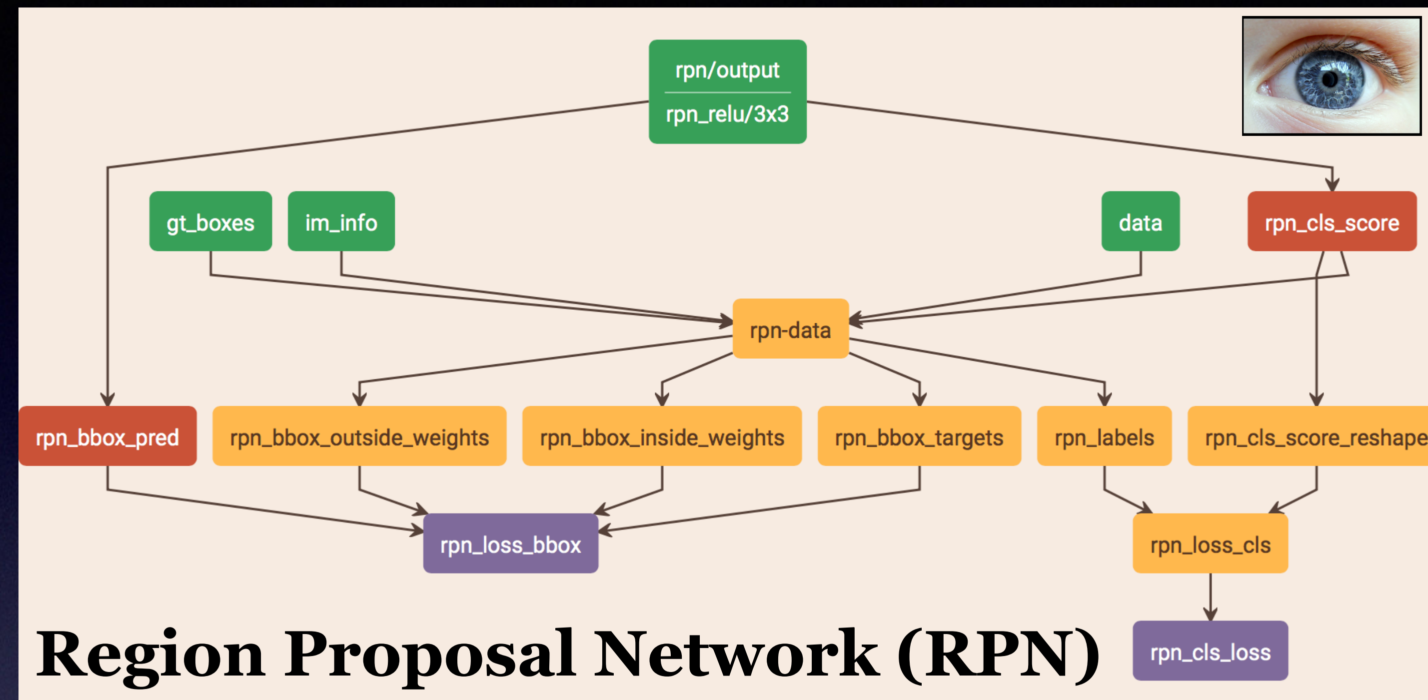


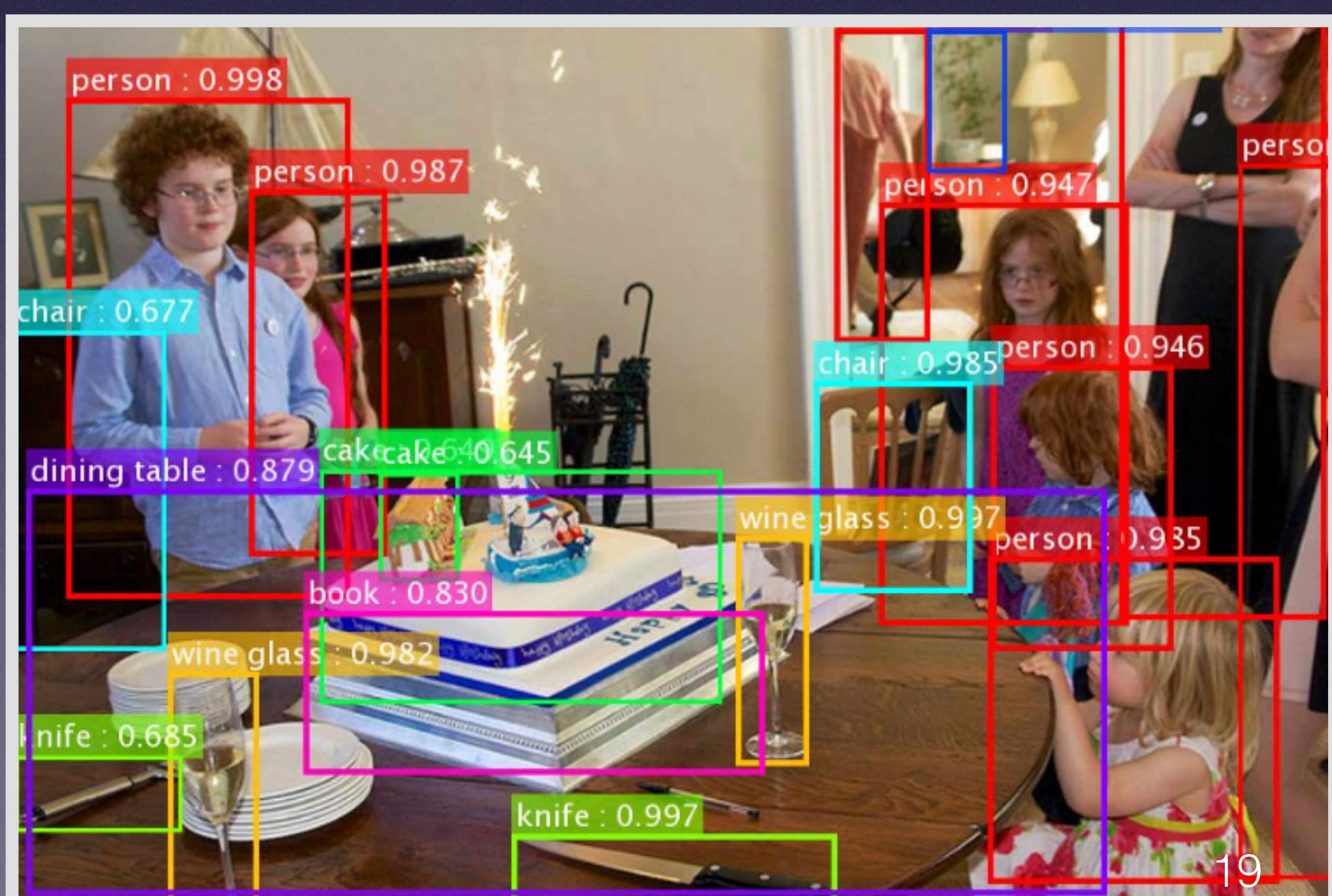
CNN for Object Detection

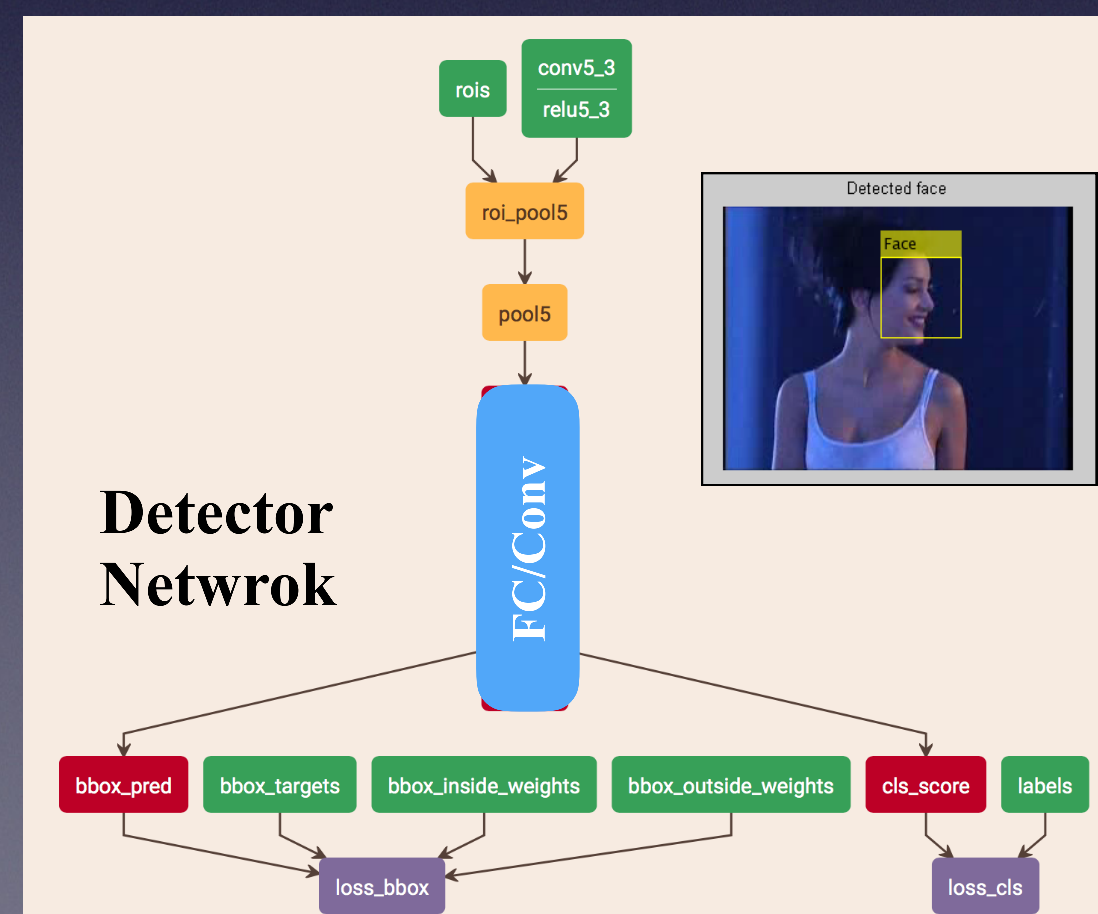
Object Detection Network

Faster-RCNN

Two sub-network to piggy-back the core classification network. Regressed to learn a bounding box with an object label







CNN for Object Detection

State-of-the-Art Accuracy (2016 ILSVRC)

Faster-RCNN + (**Inception-ResNet-v2**, **ResNet**)

Use Faster-RCNN ensembles with core network architecture ResNet and Inception-ResNet-v2, google's latest inception architecture for image classification (slightly better than Inception-v4)

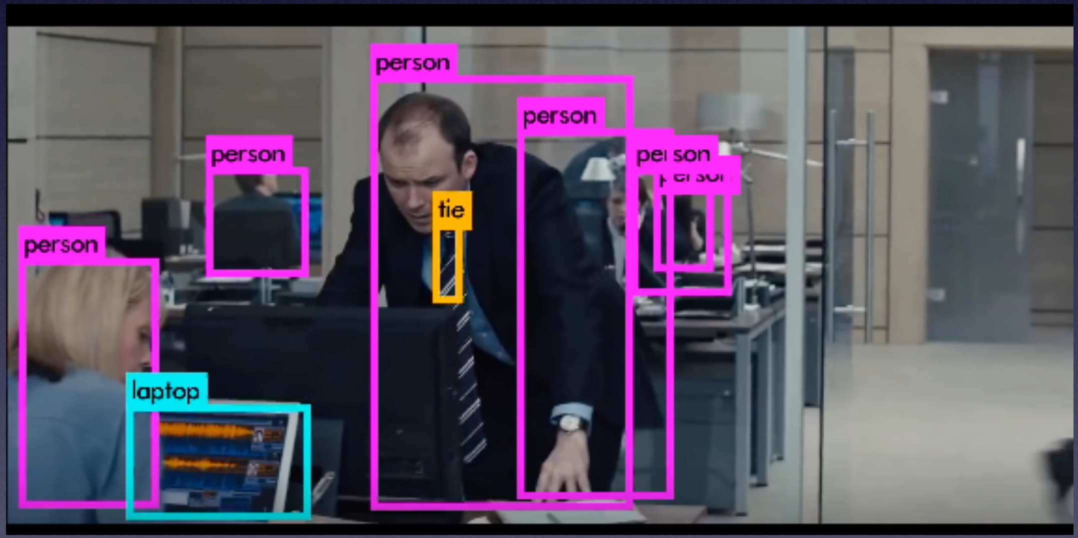
CNN for Object Detection

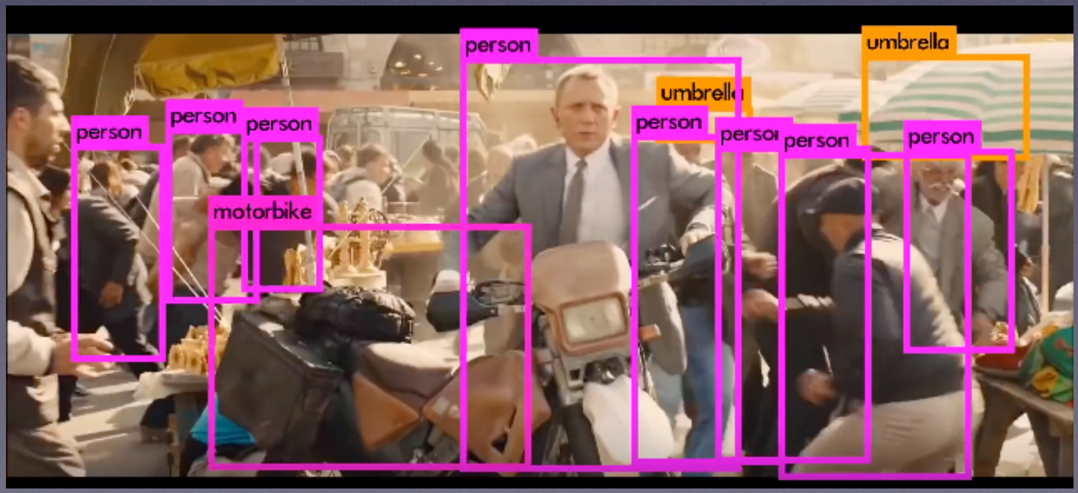
State-of-the-Art Speed

Yolo-v2

Reaches > 60 FPS processing (faster than our eyes!), author deep involved in light hardware applications (Tiny YOLO for smartphones, tablets)

Model	Train	Test	mAP	FLOPS	FPS
Old YOLO	VOC 2007+2012	2007	63.4	40.19 Bn	45
SSD300	VOC 2007+2012	2007	74.3	-	46
SSD500	VOC 2007+2012	2007	76.8	-	19
YOLOv2	VOC 2007+2012	2007	76.8	34.90 Bn	67
YOLOv2 544x544	VOC 2007+2012	2007	78.6	59.68 Bn	40
Tiny YOLO	VOC 2007+2012	2007	57.1	6.97 Bn	207
SSD300	COCO trainval	test-dev	41.2	-	46
SSD500	COCO trainval	test-dev	46.5	-	19
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40
Tiny YOLO	COCO trainval	-	-	7.07 Bn	200





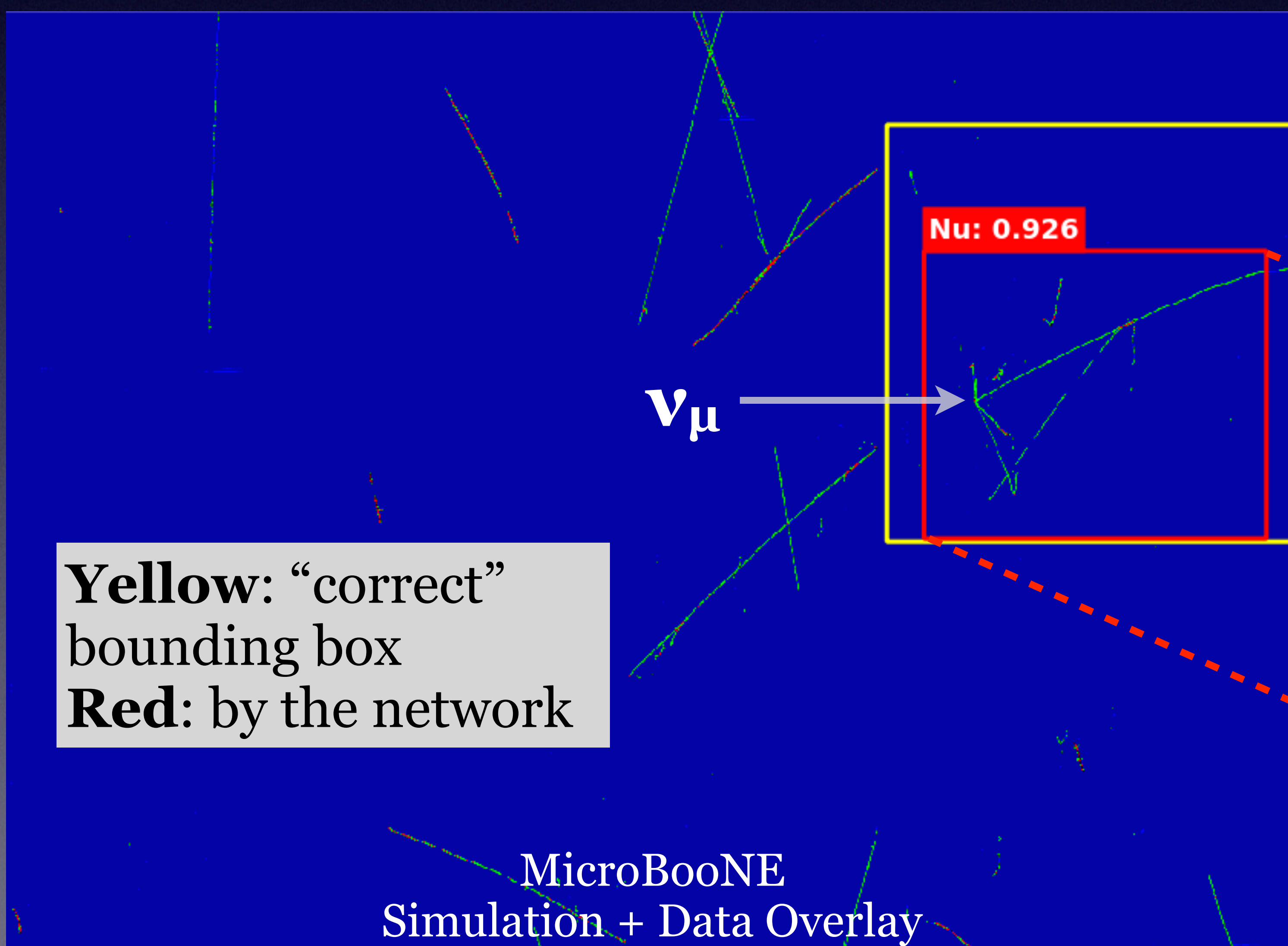
Old YOLO was a competitor for Faster-RCNN
YOLOv2 improves in both speed and accuracy

CNN for Object Detection

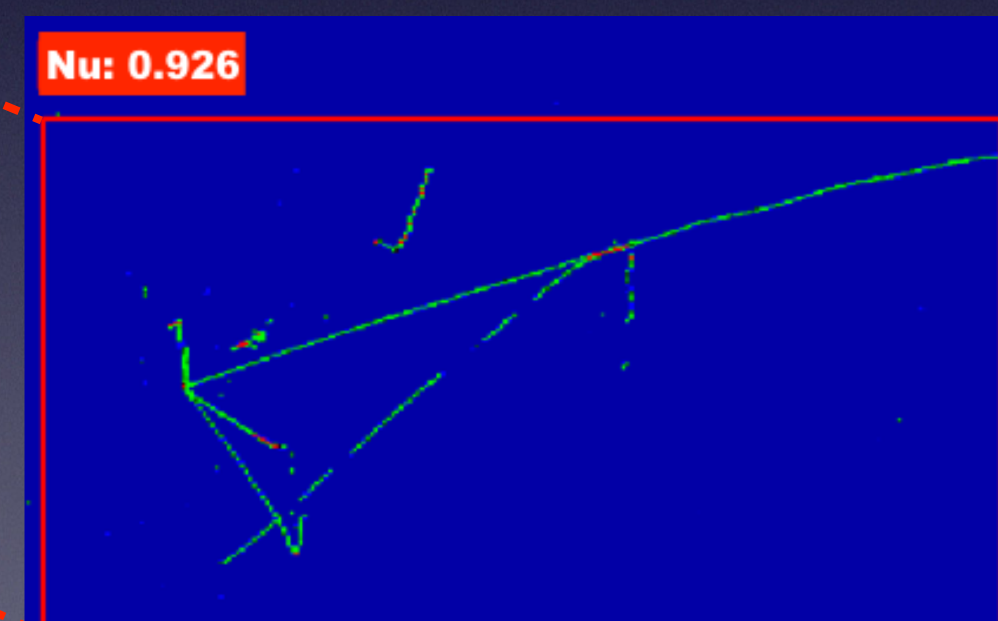
Event vertex detection

Trained a network to find neutrino interaction vertex

- Training sample uses simulated neutrino + cosmic data image
  - Supervised training using ≃101,000 collection plane images (1-plane)



arxiv:1611.05531

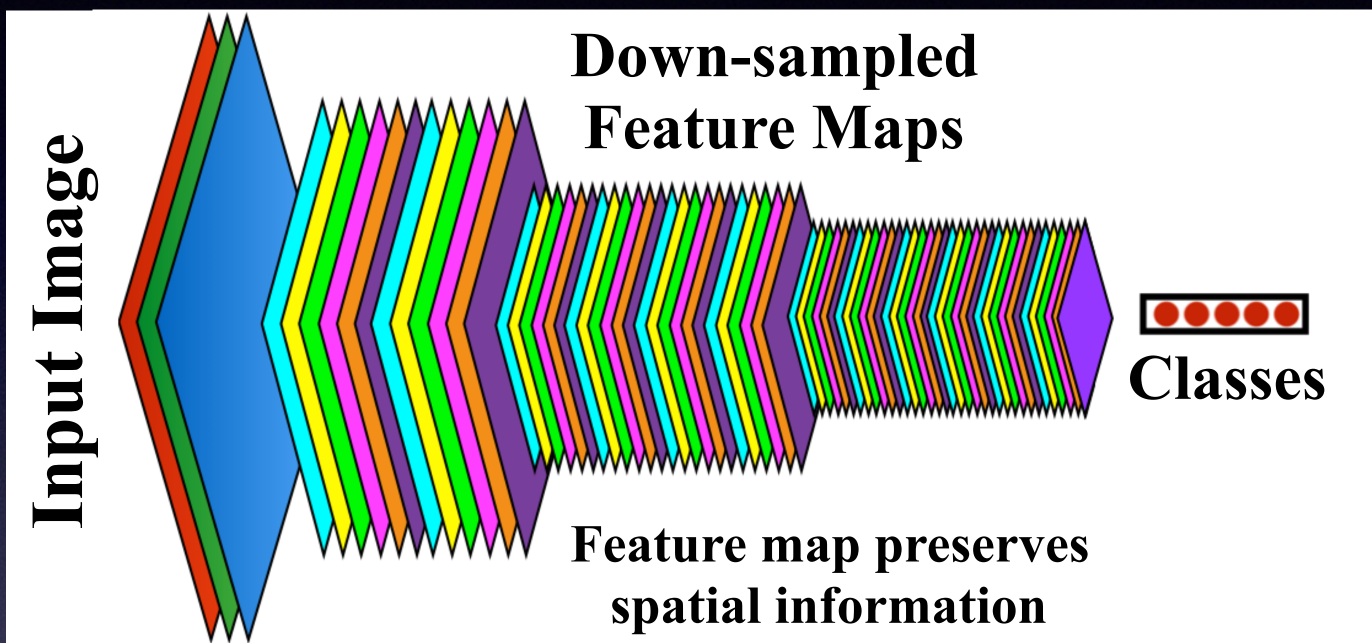


Network Output
≃ 2.6m (width) x 1 m (height)

CNN for Semantic Segmentation (SSNet)

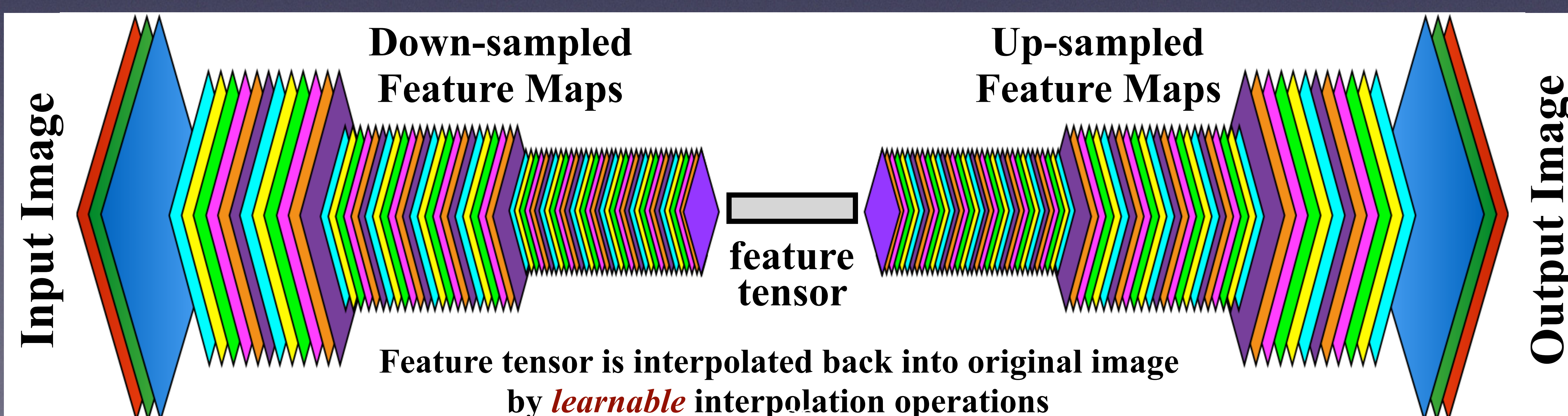
How is it different from *Image Classification*?

Example CNN for Image Classification



- Classification network reduces the whole image into final “class” 1D aray
- SSNet, after extracting class feature tensor, interpolates back into original image size

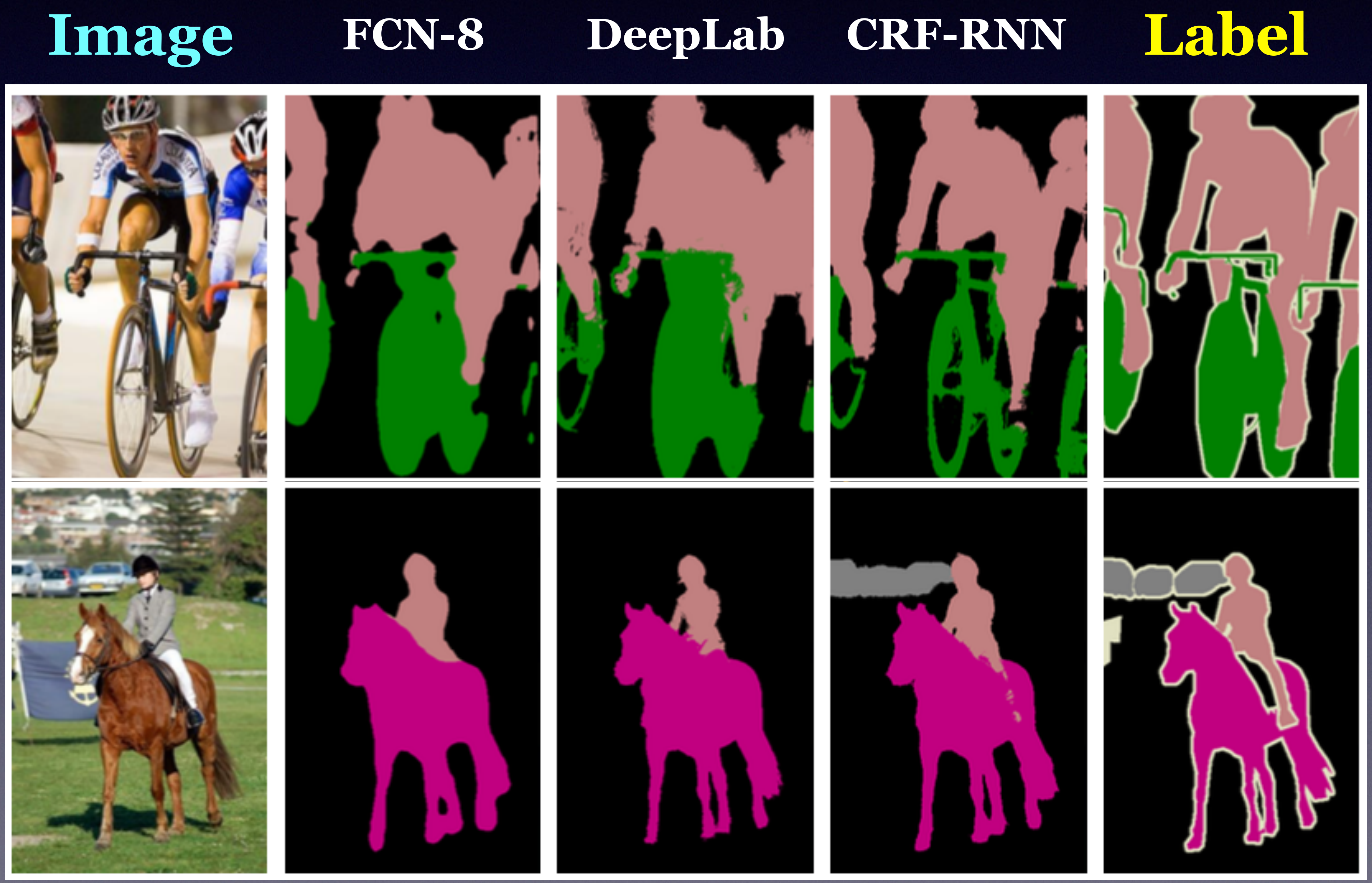
Example CNN for Semantic Segmentation



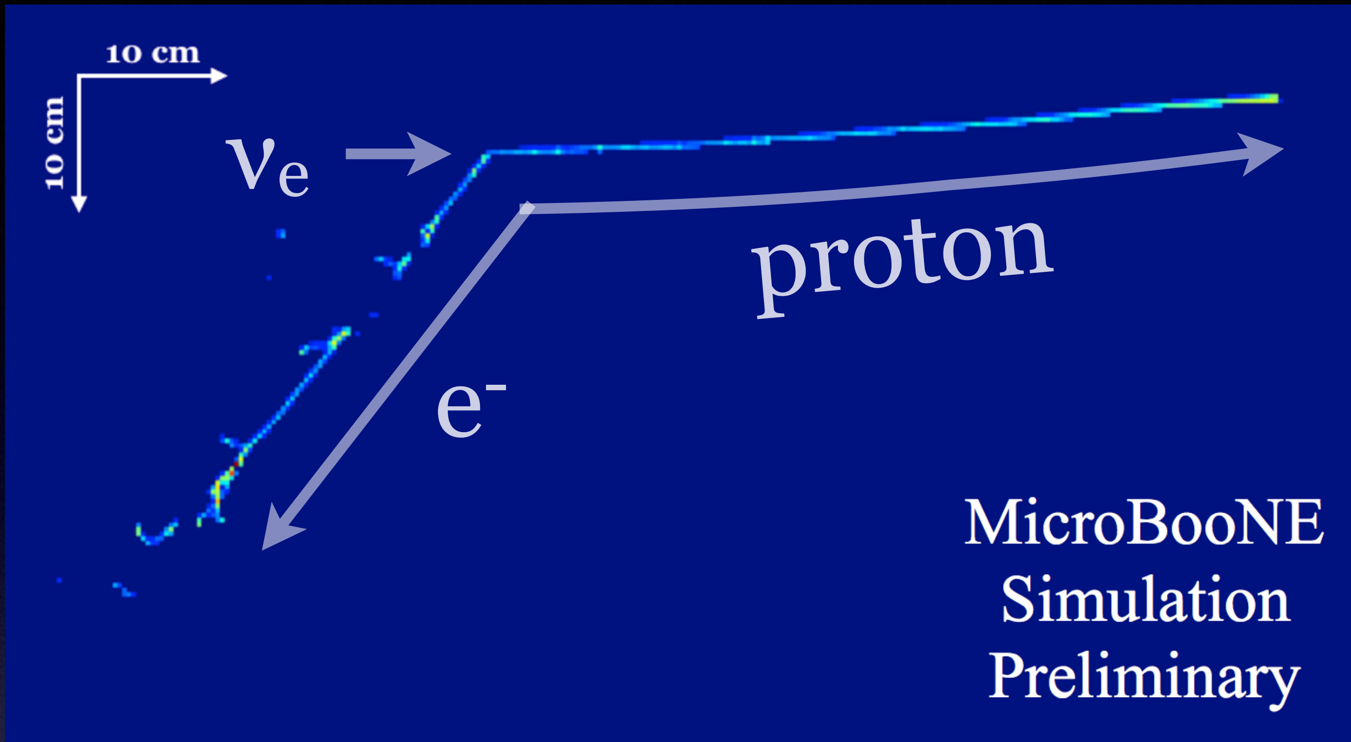
CNN for Semantic Segmentation (SSNet)

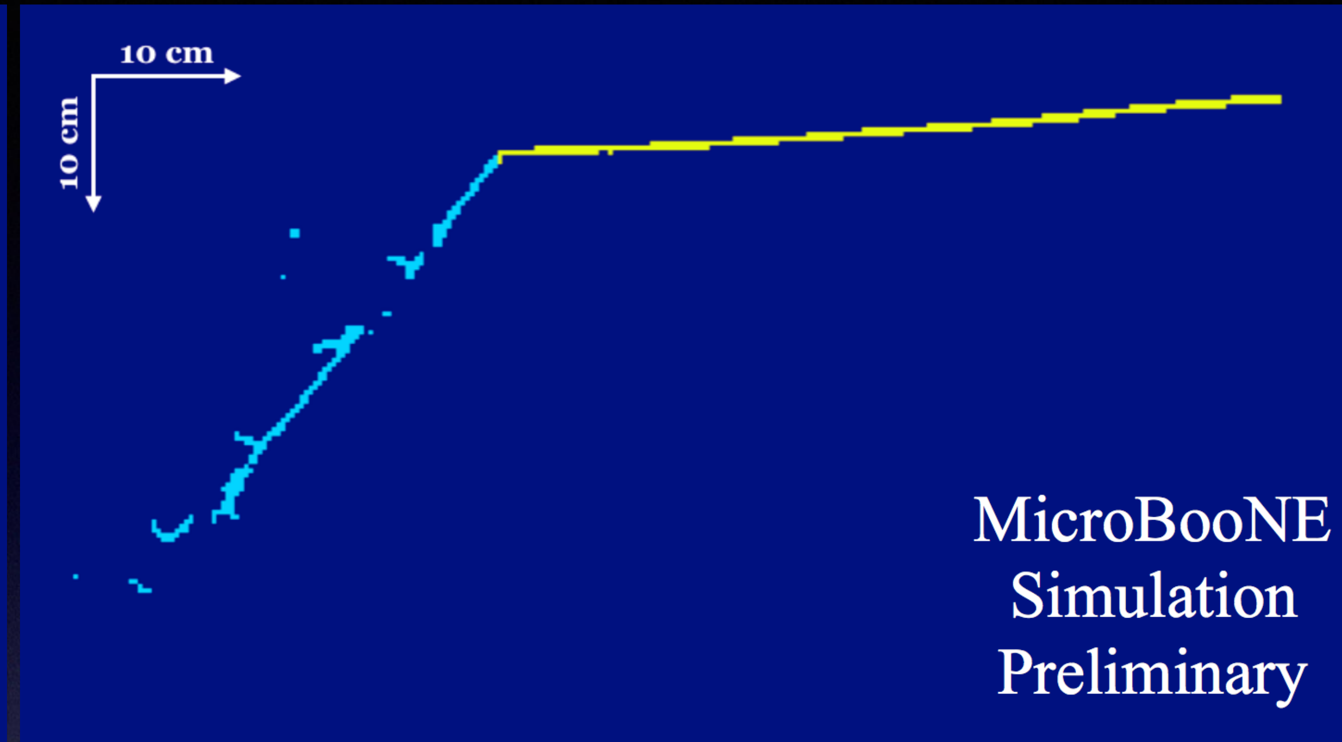
Pioneer: ***Fully-Convolutional-Network*** (FCN)

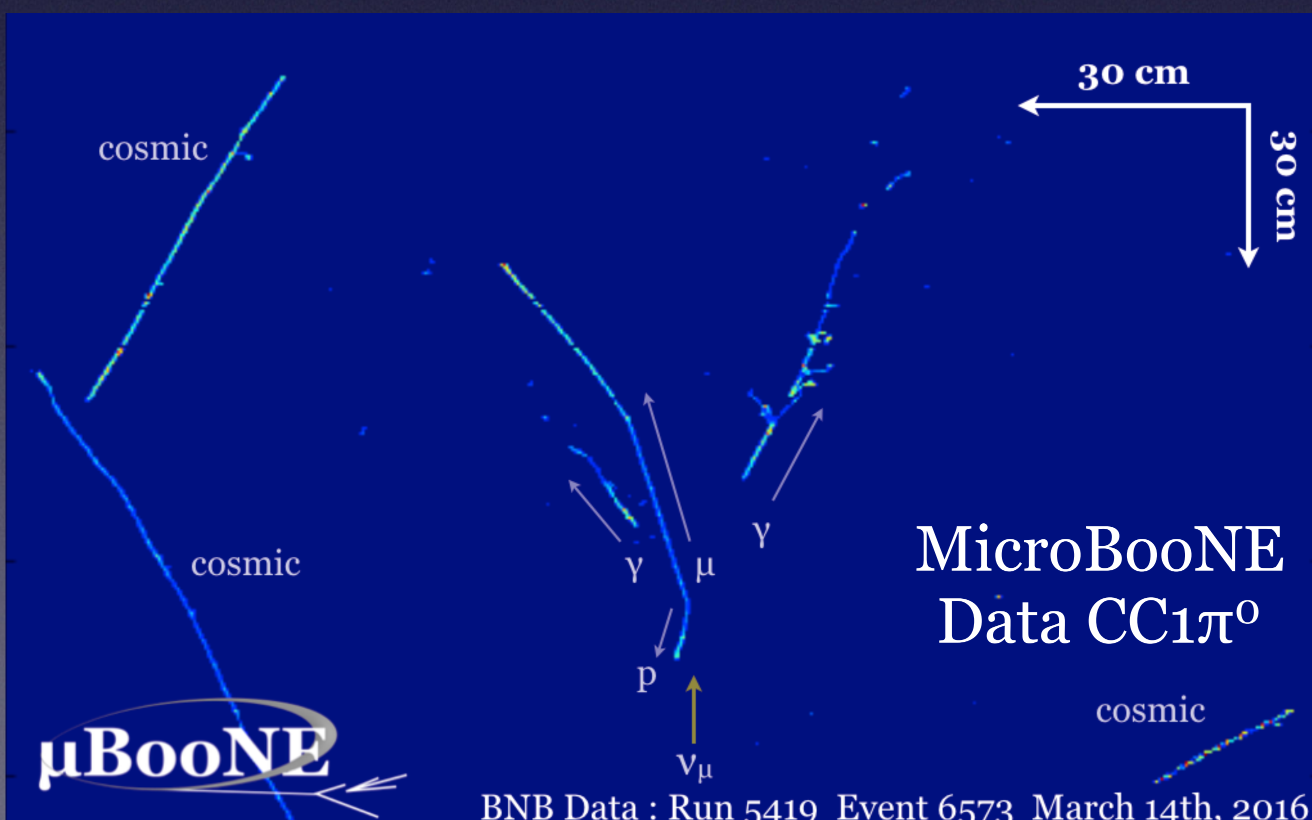
- Followed by: DeconvNet, DeepLab, CRF-RNN, SegNet, ...

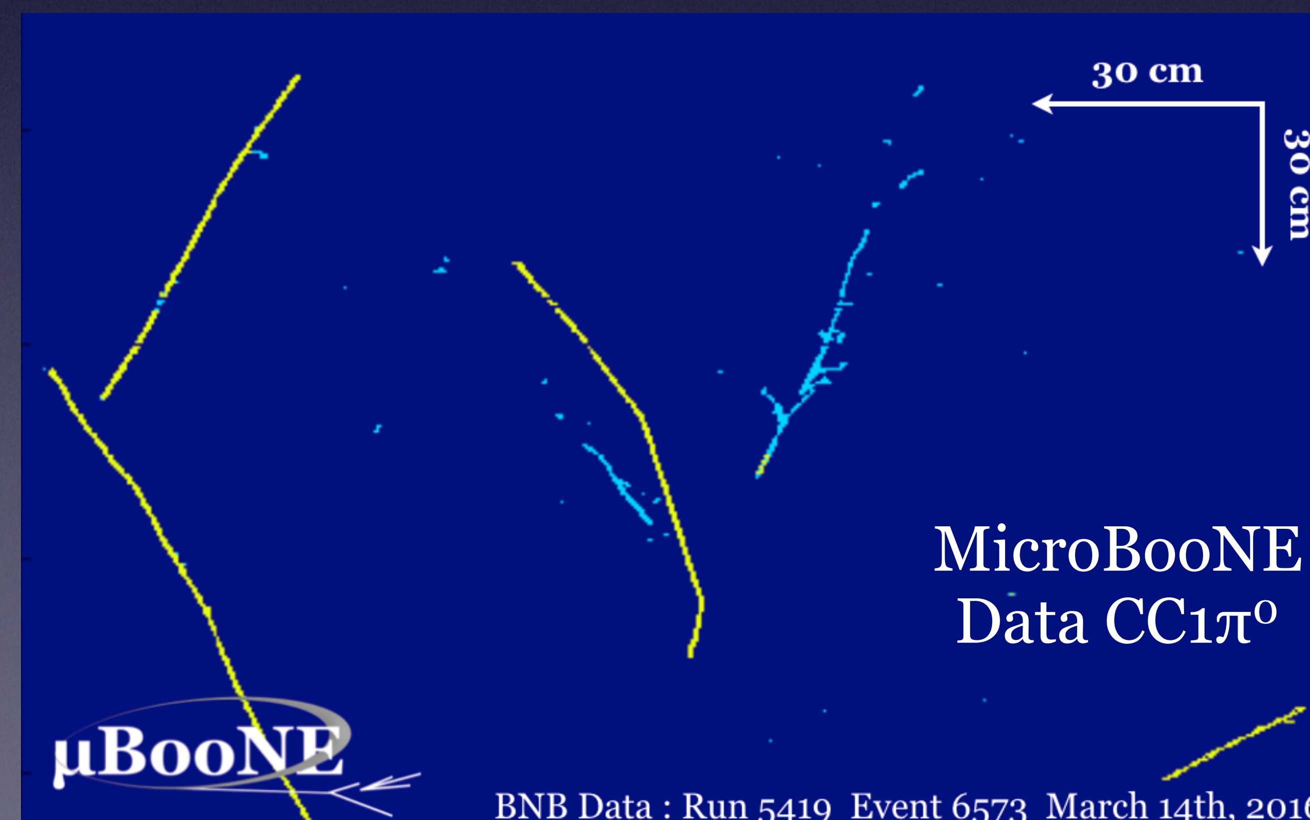


CNN for Semantic Segmentation (SSNet)









ADC Image

Network Output

CNN for Instance Semantic Segmentation (ISSNet)

State-of-the-Art Accuracy (2016 ILSVRC)

Translation-Agnostic Fully Convolutional Network

Combine RoI pooling on FCN feature maps to identify instances. Surpass performance of others that goes from an instance bounding box to pixel segmentation

... wrapping up ...

Outline

1. Introduction: naive words on how CNN works
2. Image analysis applications
3. **Summary**

My (very short) List of Papers to Highlight

Drop-out (link) … 2012
Breakthrough technique to avoid over-fitting used in AlexNet

AlexNet (link) … 2012
Legendary debut of CNN, first implementation on GPU, huge accuracy boost since last year

GoogLeNet (link) … 2014
First introduction of inception module

ResNet (link) … 2015
First introduction of residual learning

Batch-Norm. (link) … 2015
Minimize dependency on initial weights

Faster-RCNN (link) … 2015
Real-time object detection actively used to date

DC-GAN (link) … 2015
Unsupervised learning using generative adversarial architecture

FCN (link) … 2016
First fully CNN semantic segmentation

R-FCN (link) … 2016
Faster-RCNN + FCN: object detection using segmentation map

Inception-ResNet (link) … 2016
Latest inception module best performed when using ResNet

Wide-ResNet (link1, link2) … 2016
Emplirical and analytical study to show the importance of network width vs. depth

DeepLearning Softwares

Many open-source options

... + many experiment-based software interfaces

MicroBooNE has a few, too, and happy to share

- **Threaded fast IO to utilize GPU** (usually 100%)
  - Direct DL software IO interface in C++/CUDA
  - Fast numpy C-API for Python interface IO
- Various **image making algorithms**
  - 2D image classification, detection, segmentation training
  - 2D/3D Key-point feature masking
  - 3D volume data for 3D CNN feature learning
- Various **image processing algorithms**
- Qt-based 2D/3D data visualizer

Feel free to contact us if you are interested in

Wrap-Up

CNN is a limited version of fully-connected NN

- As a result, it becomes trainable to full detail data set
- Allows translational-invariant feature learning
- Suited for signal search in a homogeneous detector

CNN has a wide applications in image analysis

- Image classification, object detection, pixel labeling
- … and more not mentioned in this talk (3D, GAN, etc.)
- Thanks to a flexible, modular design of CNN architecture

Homogeneous detector neutrino experiments

- Improvement using CNNs for physics analysis
- Data reconstruction using CNNs, flexible structure allows task-by-task comparison with traditional method possible

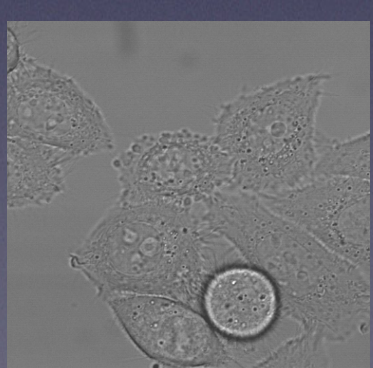
Back up

SSNet UB Analysis

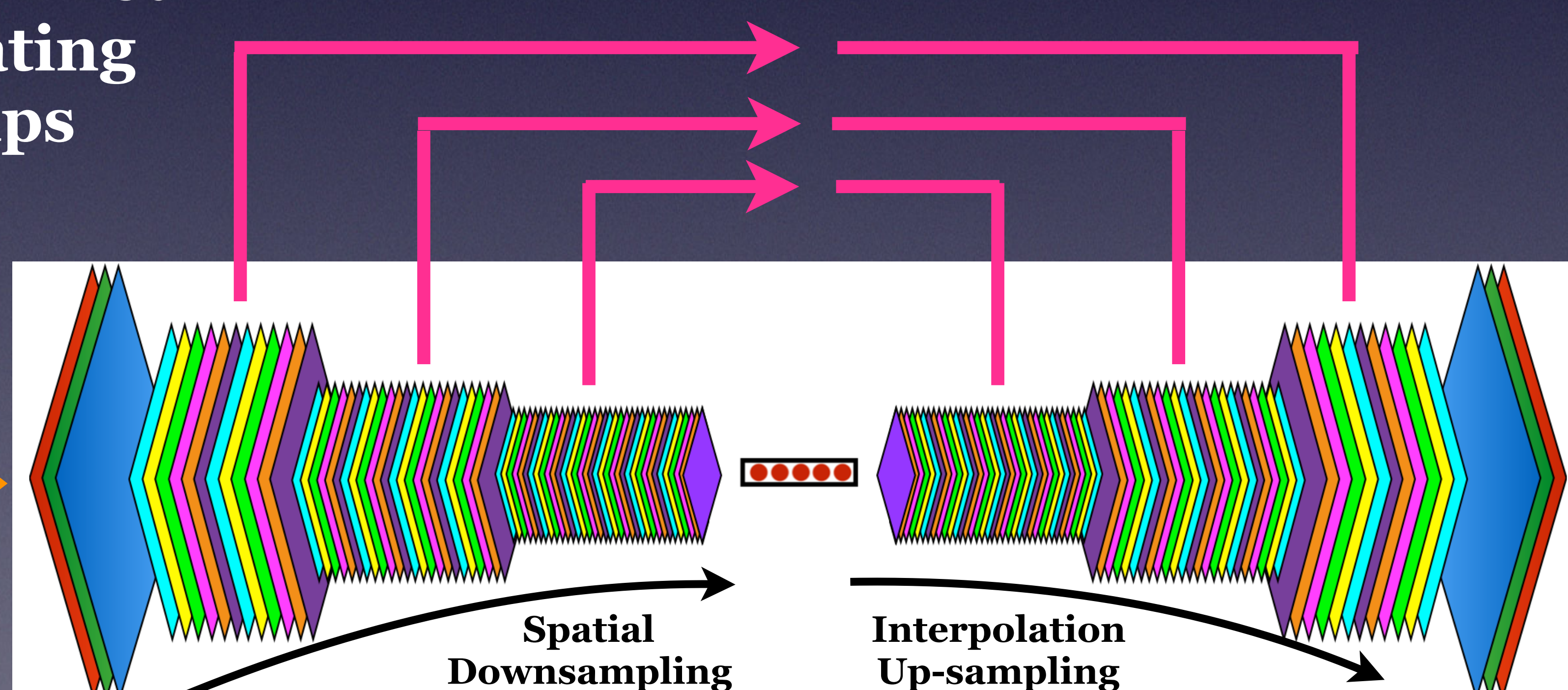
U-Net + **ResNet** module design

- Developed for bio-medical research
  - … to mask pixels of living cells (for automatized image analysis)
  - Designed for better spatial accuracy to get cells' boundary correct
- Use ResNet architecture for convolution layers

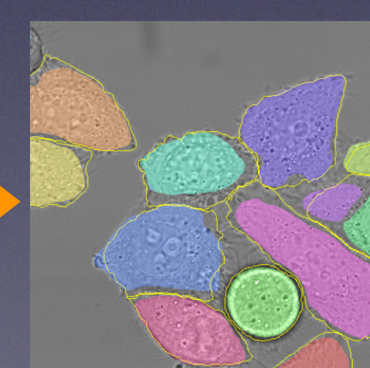
“U” shape if formed by concatenating feature maps



Biocell “Raw” Image

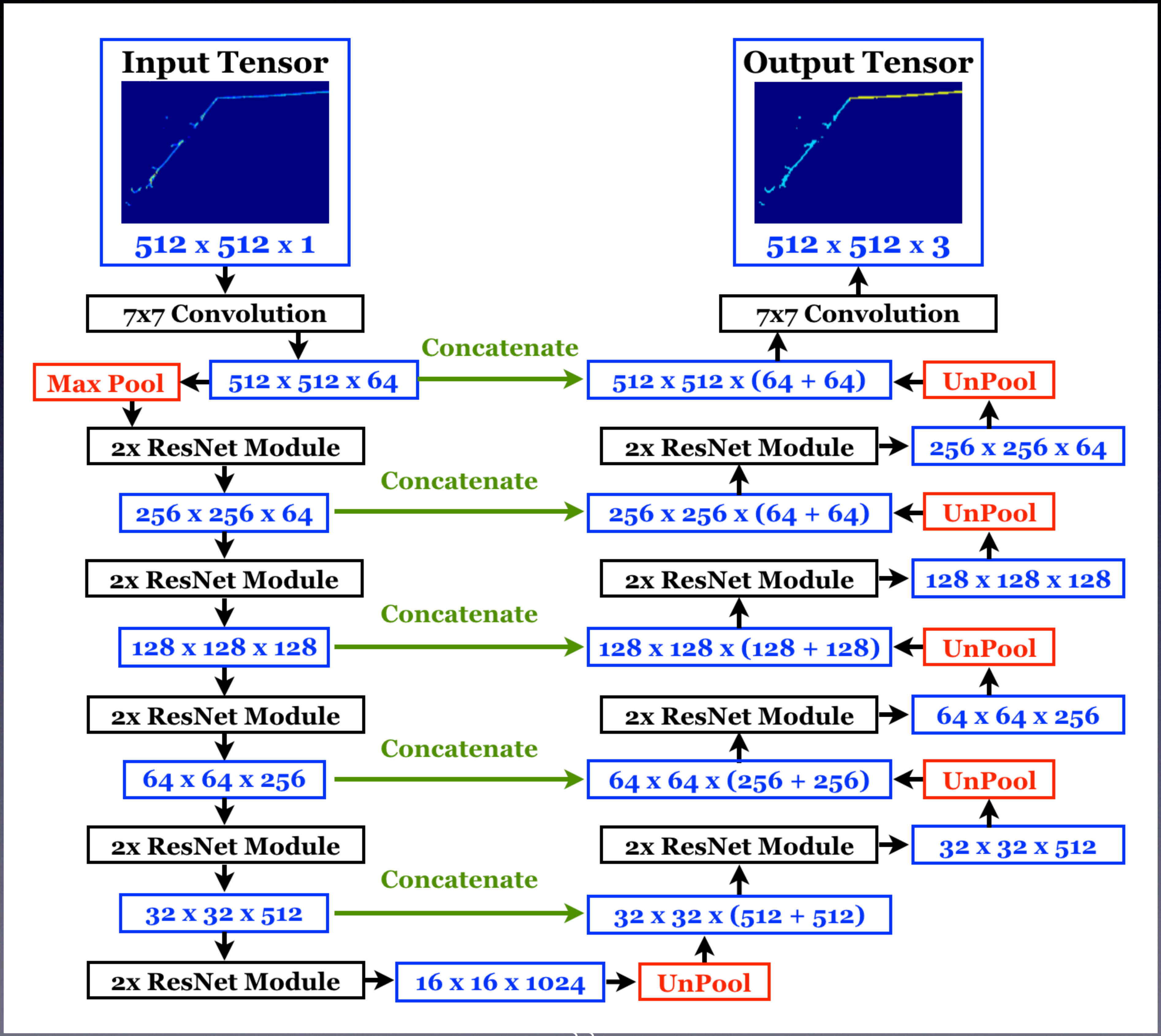


Segmented pixels of living cells (yellow boarder is truth label)
arXiv:1505.04597



Biocell Segmented

SSNet UB Analysis

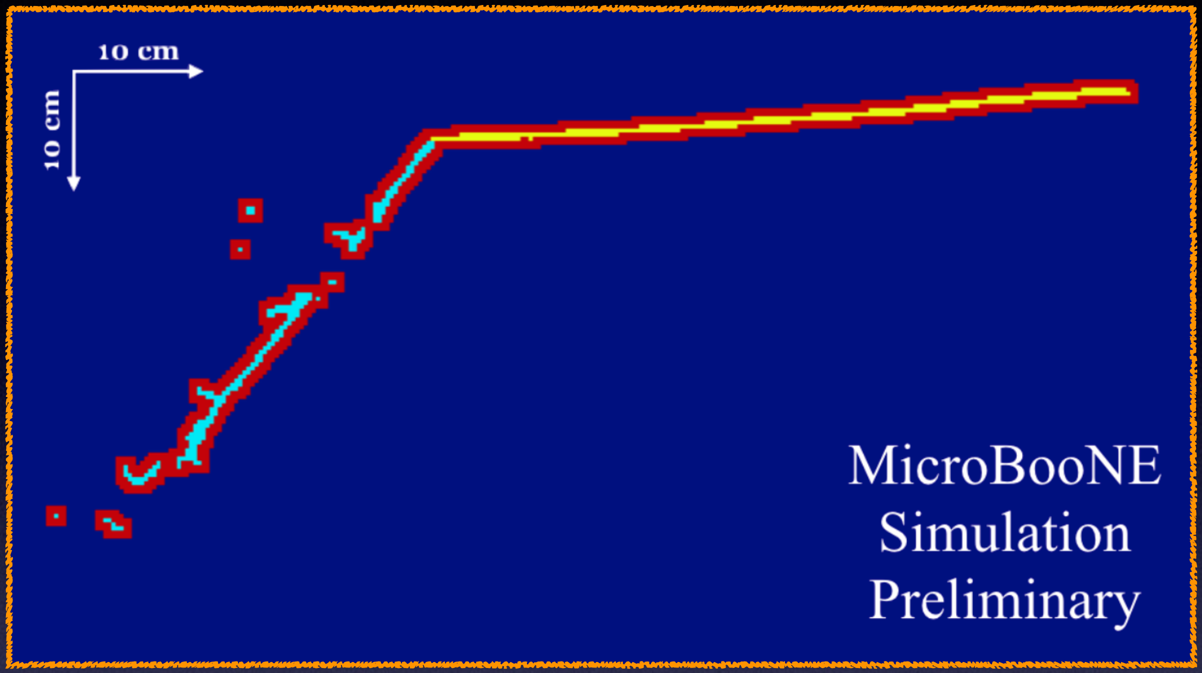


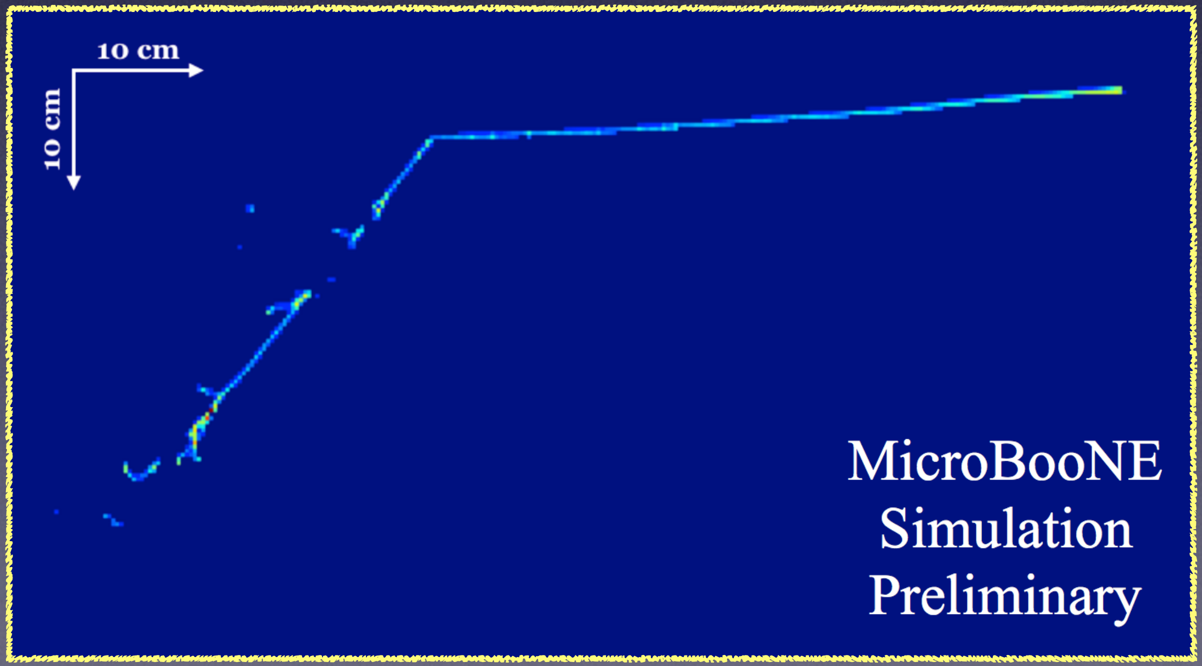
Training SSNet

"Pixel Weight" for training

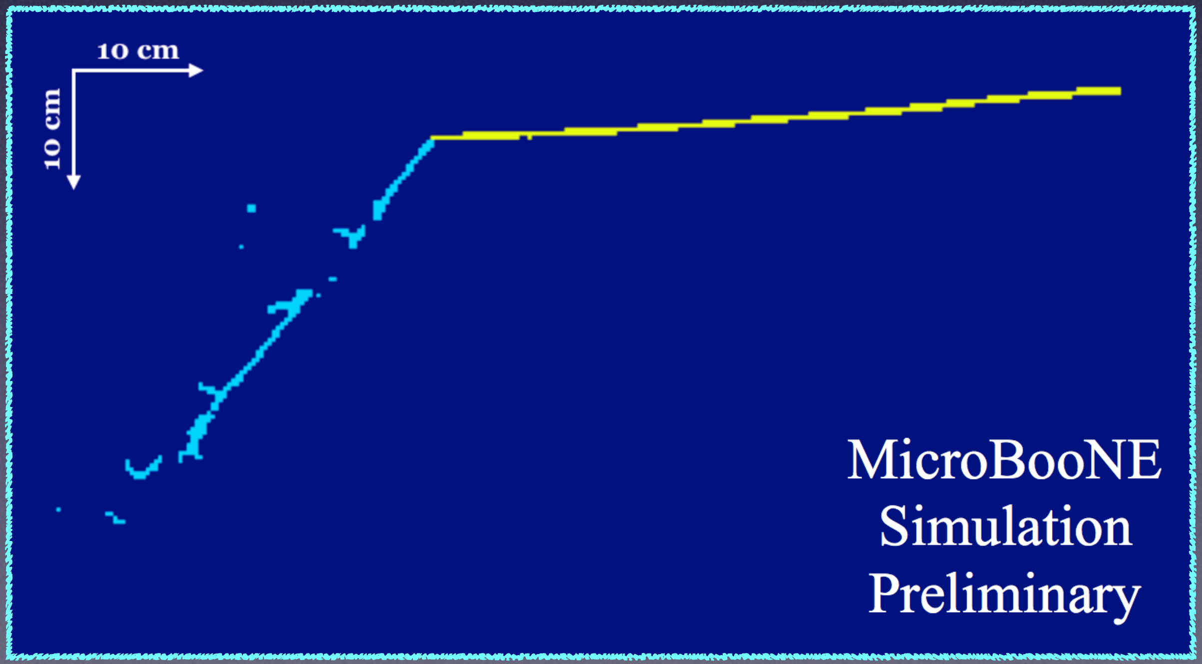
- Assign pixel-wise "weight" to penalize mistakes
- Weights inversely proportional to each type of pixel count
- Useful for LArTPC images (low information density)

"Weight" Image
(for training)





Input Image



"Label" Image
(for training)